A METHOD OF WEIL SUM
IN MULTIVARIATE QUADRATIC CRYPTOSYSTEM

A Dissertation

by

TOMOHIRO HARAYAMA

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

May 2007

Major Subject: Computer Science

A METHOD OF WEIL SUM
IN MULTIVARIATE QUADRATIC CRYPTOSYSTEM

A Dissertation

by

TOMOHIRO HARAYAMA

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Chair of Committee,	Donald K. Friesen
Committee Members,	Jianer Chen
	Bart Childs
	Maurice Rahe
Head of Department,	Valerie Taylor

May 2007

Major Subject: Computer Science

ABSTRACT

A Method of Weil Sum in Multivariate Quadratic Cryptosystem. (May 2007)

Tomohiro Harayama, B.S., Kyoto University;

M.S., Japan Advanced Institute of Science and Technology

Chair of Advisory Committee: Dr. Donald K. Friesen

A new cryptanalytic application is proposed for a number theoretic tool Weil sum to the birthday attack against multivariate quadratic trapdoor function. This new customization of the birthday attack is developed by evaluating the explicit Weil sum of the underlying univariate polynomial and the exact number of solutions of the associated bivariate equation. I designed and implemented new algorithms for computing Weil sum values so that I could explicitly identify some class of weak Dembowski-Ostrom polynomials and the equivalent forms in the multivariate quadratic trapdoor function. This customized attack, also regarded as an equation solving algorithm for the system of some special quadratic equations over finite fields, is fundamentally different from the Gröbner basis methods. The theoretical observations and experiments show that the required computational complexity of the attack on these weak polynomial instances can be asymptotically less than the square root complexity of the common birthday attack by a factor as large as $2^{n/8}$ in terms of the extension degree n of F_{2^n}. I also suggest a few open problems that any MQ-based short signature scheme must explicitly take into account for the basic design principles.

To my family.

ACKNOWLEDGMENTS

The results of this paper were obtained during my Ph.D. study at Texas A&M University. I express my deepest appreciation to my supervisor Dr. Donald K. Friesen for his generous and patient support and guidance. Without them, the successful completion of this project could never be achieved.

TABLE OF CONTENTS

CHAPTER Page

LIST OF TABLES

CHAPTER I

INTRODUCTION

A. Background and Motivation

The recent advancements of computing power and models are constantly changing the cryptographic landscapes of all the cryptographic schemes. It is now known that the Shor's quantum algorithm [1] can solve the integer factorization and discrete logarithm problems in polynomial time of input size.

This fact strongly indicates that RSA cryptosystem (Section 8.2. [2]) based on the computational hardness of integer factorization problem (Section 3.2. [2]) as well as the Diffie-Hellman schemes (Protocol 12.47. [2]) based on discrete logarithm problem (Section 3.6. [2]) are subject to the continuous erosion of the security of their underlying reference problems. And it is actually true that Elliptic Curve cryptosystems such as ECDSA or ECDH (cf. [3]) are also involved in this challenging reality. Therefore it is very important for cryptography researchers to seek more new constructions of the secure and efficient public-key cryptosystems whose reference problems are expected to remain secure even under the future advancement of computing powers and models. One such promising construction is the NP-hard problem regarding systems of *multivariate quadratic equations* over finite fields.

To see more specifically, assume that p is a prime (2 or odd) and F_q the finite field of order $q = p^n$ for $n \geq 1$. We denote $F_q^* = F_q \setminus \{0\}$. F_q is often regarded as a vector space F_p^n over F_p of dimension n with some basis. A *MQ problem* is a problem of solving n variables $x_i \in F_p$ of the randomly generated system of m quadratic equations over F_p (often $1 \leq m \leq n$) and this problem is known to be NP-

---

This dissertation follows the style of *Journal of Mathematical Cryptology*.

complete [4, 5] and hard on average as well [6]. *Multivariate quadratic cryptosystem* is a public-key cryptosystem based on the computational hardness obtained from this MQ problem (e.g., Matsumoto-Imai [7], HFE and variations: [5, 8, 9]. Tame transformation method and variations: [10, 11, 12]. Others: [13, 14]).

The MQ problem is, in fact, not rare in cryptography. For example, the computational hardness of MQ problems has many interesting combinatorial and algebraic features (cf. [15]) that are crucial for the security of the widely deployed symmetric ciphers such as Rijndael-AES [16, 17]. It is also expected that MQ problem is probably secure under the quantum computing model. In fact, there are still many potential approaches from which new types of multivariate quadratic cryptosystems might be designed.

In a multivariate quadratic cryptosystem, some trapdoor structure (denoted by $MQ(p, n, m)$-trapdoor) is designed into the public system $F(x) = (P_1(x_1, \ldots, x_n), \ldots, P_m(x_1, \ldots, x_n))$ of m quadratic polynomials in n variables over F_p so that the equation $z = F(x)$ for given $z \in F_p^n$ can be secretly inverted with the designed trapdoor information. The concrete MQ-trapdoor is actually a certain subset of the family of all the possible quadratic polynomials over F_p, whereby choosing one function in $MQ(p, n, m)$-trapdoor is presumably regarded as choosing one mapping from the set of all possible mappings $F_p^n \to F_p^m$ (cf. random mappings, Section 2.1.6 [2]). The existing trapdoor structures often consist of 3 mappings as $F = S \circ C \circ T$ with $(S, C, T) \in AL_n(F_2) \times MQ(F_2^n, F_2^m) \times AL_n(F_2)$, where $AL_n(F_2)$ and $AL_m(F_2)$ denote the set of all invertible affine mappings over F_2^n and F_2^m, resp. and $MQ(F_2^n, F_2^m)$ denotes the set of all possible systems of m quadratic polynomials in n variables over F_2.

In contrast to the well-known public-key cryptosystems such as RSA cryptosystem with parameter domain $Z/(n)$ or the Diffie-Hellman key agreement with $Z/(p)$

with prime p whose moduli must usually be assigned quite large values for practical applications (e.g. RSA: 1024-2046 bits, DH: 512-1024 bits), most of the practical multivariate quadratic cryptosystems apply characteristic p of relatively small prime (e.g.,2). As a result multivariate cryptosystems do not require any intensive use of prime number generation or primality test mechanism (Chapter 4. [2]).

One of the earliest instances of multivariate quadratic cryptosystems is the *Hidden Field Equation (HFE) cryptosystem* proposed by Patarin in 1996 [5] which is a generalization of the *Matsumoto-Imai (MI) cryptosystem* [7, 18]. The MI system uses a well-known permutation monomial $x^{2^\alpha+1}$ over F_{2^n} (cf. Theorem 7.8. [19]). Patarin extended this monomial approach to quadratic multinomial approach to gain security by sacrificing the bijectivity and simplicity in the inversion step of the central polynomials.

In comparison to many other cryptosystems, a multivariate cryptosystem usually gives a *short* signature size when applied to a digital signature scheme. For example, a digital signature scheme Quartz [20, 21] which is based on the HFE system as a trapdoor function can maintain 128-bit signature size (Section 6.1.3. [22]) even under the recent developments of cryptanalyses [23, 24, 6, 25]. With *SFLASH* [26] which uses the permutation monomial of MI system, the modified version of SFLASH v.2 [27] is still secure with the 259-bit signature size.

The quadratic portions of the HFE polynomials and the polynomials of other types of MQ schemes were known as *Dembowski-Ostrom* polynomials [28] and they have been studied in various contexts [29, 30, 31, 32, 33]. A few new classes of Dembowski-Ostrom permutation polynomials were also reported in [29]. Although it is yet to be investigated whether we can develop any cryptographic application from these new Dembowski-Ostrom permutation multinomials, one of the Dembowski-Ostrom permutation polynomials in [29] $f_\alpha(x) = xTr(x) + (\alpha+1)x^2$ over $F_{2^{kn}}$ (n

is odd, $\alpha \neq 0, 1$) has the nontrivial compositional inverse formula (Theorem 1. [34]). Since the set of Dembowski-Ostrom polynomials is not closed by the functional composition, one may need to determine further permutation polynomials through an individual investigation rather than some systematic enumeration (cf. Chapter 7. [19]). Interestingly, the Dembowski-Ostrom permutation polynomial with the inverse formula [29] is an extension of the permutation monomial used in the MI cryptosystem and their cryptographic applications should be examined by further research.

On the security of HFE polynomials, Patarin's early analysis (Sections 7.2. and 7.3. [5]) lists several classes of weak polynomial keys, and their weakness is characterized by the affine multiple properties (Example 3.55. [19]). Some of the weak polynomials in the list are special types of well-known permutation polynomials such as Dickson polynomials (Theorem 7.16. [19]) and Dobbertin polynomials [35]. If one can determine an affine q-polynomial $A(x)$ in $F_{q^n}[x]$ of $f(x)$ (i.e., affine multiple) which is divisible by $f(x)$, then all the roots of $f(x)$ reside in the set of roots of $A(x)$. Patarin generalized the previous attack on MI system [18] into this affine multiple attack in order to empirically classify the class of weak polynomials used in HFE system. To see if a given HFE polynomial $f(x)$ has the affine multiple $A(x, y)$ (where $A(x) = y$) of low Hamming weight in the exponents of y, one must actually compute the affine multiple of $f(x)$ (Section 7.3. [5]).

Many new results regarding the algebraic attacks have been published so far [36, 37, 25, 24, 38] over the developments of the symmetric cryptosystem such as AES, stream ciphers and the asymmetric cryptosystems such as multivariate schemes. One of the prominent features of these algebraic attacks is that the time complexity of attacks depends on the degree of some intermediate polynomials appearing during the computation [6]. In response to these algebraic cryptanalyses, many other variations and modifications have been developed over the past ten years (HFE and

variations: [5, 8]. Tame transformation and variations: [11, 12]. Others: [13, 14, 9]). The attacks against the specific signature schemes were also developed [39, 40, 41, 42]. The equivalent classes existing in the set of the HFE polynomials are also exploited in [43].

B. Contributions and Organizations

The *trapdoor* structure is the most important component in any MQ-based encryption or digital signature scheme. The security of MQ-trapdoors designed in the multivariate quadratic cryptosystems determine the required bit-lengths of message blocks or signatures as well as other computational resources in order to achieve the desired security level. Since the existing MQ schemes have relatively large key size, the advantages such as having short block and signatures are precious ingredients in the study of multivariate cryptosystems.

However, when a multivariate signature scheme provides a short signature size, a *birthday attack* is generally applicable to the underlying system $MQ(p, n, m)$-trapdoor $F_p^n \to F_p^m$. The common notion of this attack is that the required time complexity for randomly generated polynomials is the square root of the size of ranges, i.e, $O(p^{m/2})$ (cf. [20], Facts. 2.26, 2.27 [2]). Motivated by this particular property of short sizes for message blocks or signatures, we have investigated the security of generic MQ trapdoors and the associated MQ signature schemes under birthday attacks. In order to arrive at a new approach quite different from the currently existing ones in the literature, we develop a novel application of the theory of character and *Weil sum* of finite fields (Chapter 5. [19]) to multivariate quadratic cryptosystems.

We note that the applications of the bound results of exponential sums (including Weil sums) usually appeared in the literatures on the problems of bit-security (or

partial security) of RSA, DH, ECDH, DSA, ECDSA etc. [44, 45, 46, 47], whereby the cryptosystems rely for their security on the integer factorization or discrete logarithm. Our approach is clearly different from any other previously proposed application such as the bound results of the Weil sum to analysis of partial security. In this paper we consider the *exact* evaluation (cf. [30, 31, 32, 33]) instead of the bound approach of Weil sums and examine the *full* security of a generic MQ trapdoor function in MQ problem without assuming the availability of any partial private information. As far as we know the exact Weil sum method is the first ever developed for and applied to multivariate cryptosystem. The attack complexity is usually irrelevant to the degree of polynomials which also contrasts to the other algebraic attacks proposed so far (cf. [6]).

For $m < n$, the system does not naturally induce a univariate polynomial representation so the connection to the univariate polynomial is lost. In order to regain the connection of the system to the univariate representation, we simply assume that $MQ(2, n, m)$-trapdoor is *embedded* into $MQ(2, n, n)$-trapdoor so that the solutions found on $MQ(2, n, n)$-trapdoor are also the solutions for this embedded $MQ(2, n, m)$-trapdoor.

The structure of the dissertation is the following. In Chapter II, we prepare a number of important well-known results from elementary number theory, finite fields, polynomials, and characters. Subsequently, we provide the formal description of the systems of multivariate quadratic polynomials on which our generic cryptosystem is assumed to be constructed. We fix a MQ problem and assume that some arbitrary MQ-trapdoor is readily designed. In Chapter III, the absolute values of Weil sum of generic central polynomials are computed in $p = 2$ by a simple parity checking algorithm. I.e., we introduce an explicit Weil sum evaluation algorithm of the central polynomials which fully expresses the generic MQ problem. Since many Weil sum

methods are about the bound results of the absolute values (cf. Theorem 5.38 [19]), the algorithm obtained here is of independent interest for algorithmic number theory. The results obtained in Chapter III appeared in Cryptology ePrint Archive, Report 2006/075 as "On the Weil Sum Evaluation of Central Polynomial in Multivariate Quadratic Cryptosystem" (by Tomohiro Harayama) [48].

In Chapter IV, we work on the number of solutions of some bivariate equation associated with the Dembowski-Ostrom polynomial of form $\sum_{i=1}^{D} A_i x^{p^{s_i}+1}$. Then we relate this number of solutions to the Weil sum value of the polynomial while resolving the sign of the Weil sum values. The fact that one can obtain the exact number of the solutions of this bivariate equation will later turn out to be crucial for developing the customization of the birthday attacks.

In Chapter V, we investigate the security of the MQ-trapdoor of the Dembowski-Ostrom polynomials in Chapter IV. We introduce a new customization of the birthday attack against the Dembowski-Ostrom polynomials under some special conditions in the generic MQ-trapdoor. We theoretically characterize the weak Dembowski-Ostrom polynomials by the evaluation method of the exact number of solutions of the bivariate equations in Chapter IV. It is shown that the new attack could be asymptotically better than the attack based on the ordinary birthday problems for infinitely many possible extension degrees n. The method proposed in this chapter can be regarded as an equation solving algorithm for systems of multivariate quadratic equations, and it is fundamentally different from Gröbner basis approaches.

Finally, we provide the empirical results from our experiments in Chapter VI in order to confirm the existence of the weak Dembowski-Ostrom polynomials under the linearized binomial attack in Chapter V. Although the list of the weak polynomial instances is not exhaustive, we can naturally expect that there exist such weak polynomials regardless of the size of the extension degree n of F_{2^n}. We also discuss

the identified weak polynomials in comparison with HFE polynomials considered in Gröbner basis approaches such as [6] and suggest some open questions that any MQ-based short signature scheme must consider in their design principles. The results obtained in Chapters IV, V and VI will appear in Journal of Mathematical Cryptology as "Weil Sum for Birthday Attack in Multivariate Quadratic Cryptosystem" (by Tomohiro Harayama and Donald K. Friesen) [49]. The concluding remarks are provided in Chapter VII.

CHAPTER II

PRELIMINARIES

The purpose of this chapter is to introduce a number of the important well-known results that are repeatedly used throughout this paper.

A. Elementary Number Theory

Let $d = (\alpha, e)$ denote the greatest common divisor $\gcd(\alpha, e)$ of integers α and e.

Lemma 1 *(cf. Lemma 2.6 [30]) For integers α and e, we have:*

$$(2\alpha, e) = \begin{cases} d & \text{if } e/d \text{ is odd,} \\ 2d & \text{if } e/d \text{ is even.} \end{cases}$$

Proof Suppose e/d is odd. There is an integer $m \in Z$ such that $e = (2m+1)d$. Also for some $\alpha' \in Z$, $\alpha = \alpha' d$, thus we have $2\alpha = 2\alpha' d$. Note that α' and $(2m+1)$ are relatively prime, so are $2\alpha'$ and $(2m+1)$, therefore $(2\alpha, e) = (2\alpha' d, (2m+1)d) = d$. Suppose e/d is even. There is an integer $m \in Z$ such that $e = 2md$. While α' and $2m$ are relatively prime, so are α' and m, thus we have $(2\alpha', 2m) = 2(\alpha', m) = 2$. Therefore, $(2\alpha, e) = (2\alpha' d, 2md) = 2d(\alpha', m) = 2d$ and we obtain the desired results. □

Lemma 2 *(Lemma 2 [18]. Lemma 4 [41]). For arbitrary positive integers α and e, we have an identity:*

$$(x^\alpha - 1, x^e - 1) = x^{(\alpha, e)} - 1.$$

Proof Let $\{r_i\}_{i \geq 0}$ be the sequence of integers obtained by the Euclidean algorithm from $r_0 = e$ and $r_1 = \alpha$. If k_0 is the largest integer such that $r_{k_0} \neq 0$, then $r_{k_0} = (e, \alpha)$. Similarly, let $\{f_i(x)\}_{i \geq 0}$ be the sequence of the polynomials in $Z[x]$ obtained by the

Euclidean algorithm starting from $f_0 = x^e - 1$ and $f_1 = x^\alpha - 1$. If $r_{k-1} = q_{k-1}r_k + r_{k+1}$ for each k with $1 \le k \le k_0$, we have:

$$x^{r_{k-1}} - 1 = (x^{r_k} - 1)(x^{r_{k-1}-r_k} + \cdots + x^{r_{k-1}-q_{k-1}r_k}) + (x^{r_{k+1}} - 1),$$

i.e., $x^{r_{k+1}} - 1$ is a remainder of the division of $x^{r_{k-1}} - 1$ by $x^{r_k} - 1$ since $r_{k+1} < r_k$. Therefore,

$$f_{k_0+1}(x) = x^{r_{k_0+1}} - 1 = x^0 - 1 = 0 \text{ and } f_{k_0}(x) = x^{r_{k_0}} - 1 \neq 0,$$

so $f_{k_0}(x)$ is the greatest common divisor $(f_0(x), f_1(x))$. Hence,

$$(x^e - 1, x^\alpha - 1) = (f_0(x), f_1(x)) = f_{k_0}(x) = (x^{(e,\alpha)} - 1),$$

which is the desired result. □

Lemma 3 *(Greatest Common Divisors. Lemma 2.1 [32] and Lemma 2.6 [30]). For arbitrary positive integers α, e and $d = (\alpha, e)$, we have:*

$$(2^\alpha + 1, 2^e - 1) = \begin{cases} 1 & \text{if } e/d \text{ is odd,} \\ 2^d + 1 & \text{if } e/d \text{ is even.} \end{cases}$$

In particular, we have $(2^\alpha - 1, 2^\alpha + 1) = 1$. When p is odd, we have:

$$(p^\alpha + 1, p^e - 1) = \begin{cases} 2 & \text{if } e/d \text{ is odd,} \\ p^d + 1 & \text{if } e/d \text{ is even.} \end{cases}$$

Proof From Lemma 1 and Lemma 2,

$$(2^{2\alpha} - 1, 2^e - 1) = 2^{(2\alpha,e)} - 1 = \begin{cases} 2^d - 1 & \text{if } e/d \text{ is odd,} \\ 2^{2d} - 1 & \text{if } e/d \text{ is even.} \end{cases}$$

Assuming the greatest common divisor of two odd integers $(2^\alpha + 1, 2^\alpha - 1) \ge 3$ leads to a contradiction as $(2^\alpha + 1) - (2^\alpha - 1) = 2 < 3$. So we have $(2^\alpha + 1, 2^\alpha - 1) = 1$

and $(2^\alpha + 1, 2^d - 1) = 1$. Now,

$$\begin{aligned}(2^{2\alpha} - 1, 2^e - 1) &= (2^\alpha - 1, 2^e - 1)(2^\alpha + 1, \frac{2^e - 1}{(2^\alpha - 1, 2^e - 1)}) \\ &= (2^d - 1)(2^\alpha + 1, \frac{2^e - 1}{2^d - 1}) \\ &= (2^d - 1)(2^\alpha + 1, 2^e - 1),\end{aligned}$$

which yields the first part of this lemma. For the case p is odd, similarly from Lemma 1 and Lemma 2,

$$(p^{2\alpha} - 1, p^e - 1) = p^{(2\alpha, e)} - 1 = \begin{cases} p^d - 1 & \text{if } e/d \text{ is odd,} \\ p^{2d} - 1 & \text{if } e/d \text{ is even.} \end{cases}$$

Therefore, we have:

$$\begin{aligned}(p^{2\alpha} - 1, p^e - 1) &= (p^\alpha - 1, p^e - 1)(p^\alpha + 1, \frac{p^e - 1}{(p^\alpha - 1, p^e - 1)}) \\ &= (p^d - 1)(p^\alpha + 1, \frac{p^e - 1}{p^d - 1})\end{aligned}$$

Assuming the greatest common divisor of two even integers $(p^\alpha + 1, p^\alpha - 1) \geq 4$ leads to a contradiction as $(p^\alpha + 1) - (p^\alpha - 1) = 2 < 4$, so $(p^\alpha + 1, p^\alpha - 1) = 2$. Since $p^d - 1$ $(< p^\alpha - 1)$ is also even, we have $(p^\alpha + 1, p^d - 1) = 2$. Recall also that $(p^e - 1)/(p^d - 1) = p^{(e/d-1)d} + p^{(e/d-2)d} + \cdots + p^d + 1$. Thus,

$$(p^{2\alpha} - 1, p^e - 1) = \begin{cases} \frac{p^d - 1}{2}(p^\alpha + 1, p^e - 1) & \text{if } e/d \text{ is odd,} \\ (p^d - 1)(p^\alpha + 1, p^e - 1) & \text{if } e/d \text{ is even.} \end{cases}$$

□

Lemma 4 *(Linear Congruence Equation. cf. [50]). For arbitrary integers i, u, v and positive integer n, the equation:*

$$iu \equiv v \mod n$$

is solvable for $i \in Z$ if and only if (u, n) divides v. When the equation is solvable, there are (u, n) distinct solutions.

Proof If the equation is solvable for $i \in Z$, then there exists some integer $k \in Z$ such that $iu'(u, n) = kn'(u, n) + v$ with $u = u'(u, n)$ and $n = n'(u, n)$. So (u, n) divides v. Conversely, assume that (u, n) divides v with $v = v'(u, n)$. Then, we must equivalently check the solvability of:

$$iu' \equiv v' \mod n'.$$

Since u has the inverse $(u')^{-1} \mod n'$ for $(u', n') = 1$, we have:

$$i \equiv (u')^{-1}v' \mod n',$$

so i is obtained. If this equation is solvable, for $k \in Z$ with $v = v'(u, n)$ we have $iu' = kn' + v'$, i.e., $iu' \equiv v' \mod n'$. For all integers $l \in Z$ such that $0 \leq l < n/n'$, $i + ln'$ also satisfies $(i + ln')u' = iu' + (lu')n' \equiv v' \mod n'$. Thus, $j = i + ln'$ is also a solution of $i \equiv (u')^{-1}v' \mod n'$. For the sufficiency condition, let j is an another solution of $i \equiv (u')^{-1}v' \mod n'$. Then, we have $i \equiv j \mod n'$. Thus for some $k \in Z$ $j = i + kn'$ and we obtained the desired results. □

Finally, we recall the notion of *birthday problem* (Facts. 2.26, 2.27 [2]) which is crucial for the cryptanalysis in Chapter 5.

Definition 5 *(Stirling Number of the Second Kind. Definition 2.25 [2]). For non-negative integers m, n with $m \geq n$, the number $m^{(n)}$ is defined as $m^{(n)} = m(m - 1)(m - 2) \cdots (m - n + 1)$. The Stirling number of the second kind, denoted $\left\{ {m \atop n} \right\}$ is defined as $\left\{ {0 \atop 0} \right\} = 1$ for $m = n = 0$, and*

$$\left\{ {m \atop n} \right\} = \frac{1}{n!} \sum_{k=0}^{n} (-1)^{n-k} \binom{m}{n} k^m,$$

for nonnegative integers m, n with $m \geq n$.

The symbol $\left\{ {m \atop n} \right\}$ counts the number of ways of partitioning a set of m objects into n nonempty subsets.

Lemma 6 *(Classical Occupancy Problem Fact. 2.36. [2]). An urn has m balls numbered 1 to m. Suppose that n balls are drawn from the urn one at a time, with replacement, and their numbers are listed. The probability that exactly t different balls have been drawn is:*

$$P_1(m, n, t) = \left\{ {m \atop n} \right\} \frac{m^{(t)}}{m^n},$$

for $1 \leq t \leq n$.

Proof Let us denote the set of n picked balls $\{i_1, i_2, \ldots, i_n\}$ in order. There are $\left\{ {n \atop t} \right\}$ possible ways to partition of n balls into t nonempty subsets. There are $m^{(t)}$ possible choices of t distinctive numbers of balls out of m numbers. Thus, we have in total $m^{(t)} \times \left\{ {n \atop t} \right\}$ possible ways to have exactly t different balls among n balls $\{i_1, i_2, \ldots, i_n\}$ drawn from the urn of m balls one at a time, with replacement meanwhile there are m^n possible ways to pick up n balls out of m balls with replacement. Therefore, we obtain the desired result:

$$P_1(m, n, t) = m^{(t)} \left\{ {m \atop n} \right\} \times \frac{1}{m^n}.$$

□

Definition 7 *(Birthday Problem. Fact 2.27. [2]) An urn has q balls numbered 1 to q. Suppose that r balls are drawn from the urn one at a time, with replacement, and their numbers are listed. Then, the probability of at least one coincidence (i.e., a ball drawn at least twice) is:*

$$P_2(q, r) = 1 - P_1(q, r, r) = 1 - \frac{q^{(r)}}{q^r},$$

for $1 \leq r \leq q$. *If* $r = O(\sqrt{q})$, *then we have:*

$$P_2(q, r) \to 1 - \exp(-\frac{r(r-1)}{2q} + O(\frac{1}{\sqrt{q}})) \approx 1 - \exp(-\frac{r^2}{2q}).$$

The probability distribution defined by $P_2(q, r)$ *is called a birthday surprise or paradox.*

It is well known that the probability that at least 2 people in a room of 23 people have the same birthday is $P_2(365, 23) \approx 0.507$, which is surprisingly large. For a fixed q, the quantity of $P_2(q, r)$ rapidly increases as r increases. In Chapter 5, we use the fact that the expected number of draws before a coincidence is $\sqrt{\frac{\pi}{2}q}$ as $q \to \infty$.

B. Finite Fields

Let p be a prime. Then the *Galois field* F_p is usually identified with $Z/(p)$ the ring of residue classes of the integers modulo a *principal ideal* generated by p. From the theorem of *Existence and Uniqueness of Finite Fields* (Theorem 2.5 [19]), any finite field has some prime power order $q = p^n$ and conversely for any prime p and $n \geq 1$, there exists a finite field of order $q = p^n$ uniquely determined up to field isomorphisms. We denote the finite extension field of F_p of degree n by F_q. The extension field F_q is often regarded as a vector space F_p^n over F_p of dimension n with some basis. Since we identify F_p with $Z/(p)$, the image of *absolute trace function* of any $x \in F_q$ are merely the integers in $[0, p-1]$.

Let F_q be a finite field. Then the set of all nonzero elements of F_q forms a *cyclic group* in terms of the multiplication operation of F_q. Denote the group by F_q^*. It is easily shown that for any $a \in F_q$, $a^q = a$.

Theorem 8 *(Power of p. Theorem 1.46. [19]). Let* F_q *be a finite field of characteristic p. For any integer* $t \in Z$ *and* $x, y \in F_q$*:*

1. $(x + y)^{p^t} = x^{p^t} + y^{p^t}$ *and*

2. $(x-y)^{p^t} = x^{p^t} - y^{p^t}$.

Proof From

$$\binom{p}{i} = \frac{p(p-1)\cdots(p-1+1)}{1\cdot 2\cdots i} \equiv 0 \mod p,$$

for $1 \leq i \leq p-1$, we have:

$$(x+y)^p = x^p + \sum_{i=1}^{p-1} \binom{p}{i} x^i y^{p-i} + y^p = x^p + y^p,$$

as the fact that ring F_q is of characteristic p turns all the intermediate terms $\binom{p}{i}x^i y^{p-i}$ of nonzero coefficients into zeros. Thus, the induction on t yields the first identity. The second identity follows from the first one, i.e.,

$$x^{p^t} = ((x-y)+y)^{p^t} = (x-y)^{p^t} + y^{p^t}.$$

□

Definition 9 *(Primitive Element. Definition 2.9. [19]). Let F_q be a finite fields. A generator of the cyclic group F_q^* of F_q is called a primitive element of F_q.*

It is well known that the primitive elements exist in any finite field F_q and serves as a *defining element* over F_p.

Definition 10 *(Conjugate Element. Definition 2.18. [19]). Let F_q be a finite field, which is an extension of F_p where $q = p^n$. For $a \in F_q$, the elements $a^p, a^{p^2}, \ldots, a^{p^{n-1}}$ are called the conjugates of a with respect to F_q.*

By automorphisms σ of F_q we mean an automorphism of F_q that fixes the elements of F_p. For $0 \leq i \leq n-1$:

$$\sigma_i(a) = a^{p^i},$$

are also called *Frobenius mappings* (cf. Theorem 2.21. [19]). For any Frobenius mapping $\sigma_i(a) = a^{p^i}$ we often use the fact that $\sigma_i(a)$ runs throughout F_q as a runs

throughout F_q in the various passages of the proofs of Weil sum methods in the later chapters.

The theory of polynomials over finite fields is the main mathematical knowledge necessary for designing and analyzing the currently proposed multivariate quadratic cryptosystems. The following polynomial called *linearized polynomial* has very important features in our cryptosystems.

Definition 11 *(Linearized Polynomial. Definition 3.49. [19]) A polynomial of the form:*

$$L(x) = \sum_{i=0}^{m-1} b_i x^{q^i},$$

with coefficients in an extension field F_{q^m} is called a q-polynomial over F_{q^m}.

When the value of q is fixed once and for all or is clear from the context, it is called a *linearized polynomial*. In this paper, we only consider a linearized polynomial as p-polynomial over F_q. As is easily shown, we have:

$$L(x+y) = L(x) + L(y) \text{ and } L(ax) = aL(x),$$

for all $a \in F_q$ and $x \in F_{q^m}$. Thus, a linearized polynomial is considered a linear mapping over F_{q^m} when we regard F_{q^m} as as F_q-vector space. Suppose we only consider a p-polynomial over F_q by regarding F_q as a vector space F_p^n of dimension n over F_p. If we set the corresponding $n \times n$ matrix B_L over F_p to this L, it is easy to solve the equation $y = L(x)$ by applying the *Gaussian Elimination* on the matrix B_L. In other words, we can easily obtain the inverse B_L^{-1} which corresponds to the inverse linearized polynomial $L^{-1}(x)$ over F_q if $L(x)$ is bijective.

A *trace function* $Tr_t : F_q \to F_{p^t}$ for some integer $t \geq 1$ which divides n, is defined by

$$Tr_t(x) = x + x^{p^t} + x^{p^{2t}} + \ldots + x^{p^{(n/t-1)t}},$$

for all $x \in F_q$ and the *absolute trace function* is simply denoted by Tr (when $t = 1$).

Theorem 12 *(Properties of Trace. Theorem 2.23. [19]). Let $q = p^n$ and $t \in Z$ a positive integer. For all $x, y \in F_q$ and $a \in F_{p^t}$, we have:*

1. $Tr_t(ax) = aTr_t(x)$,
2. $Tr_t(x + y) = Tr_t(x) + Tr_t(y)$ *and*
3. $Tr_t(x^{p^t}) = Tr_t(x)$.

Proof Since $a^{p^t} = a$ for all $a \in F_{p^t}$, we have:

$$\begin{aligned} Tr_t(ax) &= (ax) + (ax)^{p^t} + \cdots + (ax)^{p^{(n/t-1)t}} \\ &= ax + ax^{p^t} + \cdots + ax^{p^{(n/t-1)t}} \\ &= aTr_t(x). \end{aligned}$$

By using Theorem 8, we have:

$$\begin{aligned} Tr_t(x + y) &= (x + y) + (x + y)^{p^t} + \cdots + (x + y)^{p^{(n/t-1)t}} \\ &= x + y + x^{p^t} + y^{p^t} + \cdots + y^{p^{(n/t-1)t}} + x^{p^{(n/t-1)t}} \\ &= Tr_t(x) + Tr_t(y). \end{aligned}$$

Therefore we have:

$$\begin{aligned} Tr_t(x^{p^t}) &= (x^{p^t}) + (x^{p^t})^{p^t} + \cdots + (x^{p^t})^{p^{(n/t-1)t}} \\ &= (x^{p^t}) + (x^{p^t})^{p^t} + \cdots + (x^{p^t})^{p^n} \\ &= Tr_t(x). \end{aligned}$$

□

A *additive character* χ of the additive group of F_q is a homomorphism from F_q to the multiplicative group U of complex numbers of absolute value 1. That is, χ is

a mapping from F_q to U with $\chi(x_1 + x_2) = \chi(x_1)\chi(x_2)$ for all $x_1, x_2 \in F_q$ (Section 1 [19]). The character is called *canonical additive character* if $\chi_1(x) = \exp(2\pi i Tr(x)/p)$ for $x \in F_q$. From Theorem 5.7 [19], any additive character χ_a of F_q is obtained from

$$\chi_a(x) = \chi_1(ax),$$

for all $x \in F_q$ with some $a \in F_q$. The following properties will turn out to be very useful in the later chapters.

Theorem 13 *(Properties of Additive Character. Chapter 5. [19]). For all $x, y \in F_q$, we have:*

1. $\chi_1(x + y) = \chi_1(x)\chi_1(y)$ *and*
2. $\chi_1(x^p) = \chi_1(x)$.

Proof From Theorem 12 we have $Tr(x + y) = Tr(x) + Tr(y)$. Thus,

$$\exp(2\pi i Tr(x + y)/p) = \exp(2\pi i Tr(x)/p) \exp(2\pi i Tr(y)/p).$$

Similarly, since $Tr(x^p) = Tr(x)$, we also have

$$\exp(2\pi i Tr(x^p)/p) = \exp(2\pi i Tr(x)/p),$$

and we obtained the desired results. □

Let $F_q^\wedge$ denote the set of all additive characters of F_q.

Theorem 14 *(Properties of Character Sum. Theorem 5.4. [19]). For any nontrivial additive character $\chi \in F_q^\wedge$, we have:*

$$\sum_{x \in F_q} \chi(x) = 0.$$

Also for any $x \neq 0 \in F_q$ we have:

$$\sum_{\chi \in F_q^\wedge} \chi(x) = 0.$$

Proof Since χ is nontrivial, there exists $a \neq 0 \in F_q$ such that $\chi(a) \neq 1$. Thus,

$$\chi(a) \sum_{x \in F_q} \chi(x) = \sum_{x \in F_q} \chi(a+x) = \sum_{a+x \in F_q} \chi(a+x) = \sum_{x \in F_q} \chi(x).$$

Therefore, we have:

$$(\chi(a) - 1) \sum_{x \in F_q} \chi(x) = 0,$$

which implies the first part of the theorem. For the second part, we denote the function $\widehat{a}$ defined as:

$$\widehat{a}(\chi) = \chi(a),$$

for all $\chi \in F_q^\wedge$ and $a \in F_q$ It is an easy matter to show that $F_q^\wedge$ forms an abelian group with addition defined as:

$$(\chi_a + \chi_b)(x) = \chi_a(x)\chi_b(x)$$

for χ_a, χ_b: additive characters and $x \in F_q$ with the trivial additive character χ_0 as the identity element. Therefore we have:

$$\sum_{\chi \in F_q^\wedge} \chi(a) = \sum_{\chi \in F_q^\wedge} \widehat{a}(\chi) = 0.$$

□

Now, we are ready for introducing a *Weil sum* of polynomials over finite fields.

Definition 15 *(Weil Sum. Chapter 4. [19]). Let χ be a nontrivial additive character*

of F_q and let a polynomial $f(x) \in F_q[x]$ be of positive degree. Then, the sum:

$$\sum_{x \in F_q} \chi(f(x)),$$

is called a Weil sum of $f(x)$.

As a final preparation of finite fields, we recall a result regarding the solvability of equations over F_q.

Corollary 16 *(Solvability. Theorem 3.1. [32]). Let g be a primitive element of F_q. Let α be an integer and $a = g^s$ be a nonzero element of F_q with some integer s. Then, the finite field equation:*

$$x^\alpha = a,$$

is solvable for $x \in F_q$ if and only if $(\alpha, q-1)$ divides s.

Proof Expressing $x \in F_q$ of the equation by the conjugate of primitive element g as $x = g^r$ for some (unknown) integer s, we want to know the solvability of the equation:

$$g^{r\alpha} = g^s,$$

for $r \in Z$. Equivalently, we need to know the solvability of the associated linear congruence equation in the exponents:

$$r\alpha \equiv s \mod q-1,$$

for $r \in Z$. From Lemma 4, this equation is solvable if and only if $(\alpha, q-1)$ divides s, which is a desired result. □

C. Multivariate Quadratic Cryptosystems

Let q be some power of p. We fix two finite fields F_q and its extension F_{q^n}. In general, a function $F : X \to Z$ is called a *oneway function* if it is "easy" to compute $z = F(x)$ for all $x \in X$ but for essentially all elements $z \in Im(F)$, the image by F, it is computationally infeasible (i.e., not in polynomial time) to find any $x \in X$ such that $F(x) = z$. It is well known that [2]:

$$\text{Existence of oneway function } \Rightarrow \mathcal{P} \neq \mathcal{NP}.$$

In order to build public key encryptions and digital signature schemes with any kind of candidates of oneway functions, we need one more additional feature in the onewayness of these functions.

Definition 17 *(Trapdoor Oneway Function. Definition 1.16. [2]). A trapdoor oneway function is a oneway function $F : X \to Z$ with the additional property that given some extra information called trapdoor information it becomes feasible to find, for any given $z \in Im(F)$, an $x \in X$ such that $F(x) = z$.*

Since it is still unknown if there exists a oneway function in a rigorous sense, it is also unknown if there exists a trapdoor oneway function. However, there are a number of good candidates for oneway and trapdoor oneway functions. We will introduce one of the currently known candidates for a oneway function and for a trapdoor oneway function in the following.

1. MQ Problem and Representations of Polynomials

Oneway functions and trapdoor oneway functions often called the *cryptographic reference problems.* This is because the security of many public-key cryptosystems relies

on the apparent intractability of onewayness obtain from these functions. We now consider one such cryptographic reference problem consisting of the systems of multivariate quadratic polynomials over finite fields.

Definition 18 *(MQ Problem. cf. [25, 20]). Let* $P_1, \ldots, P_m \in F_q[x_1, \ldots, x_n]$ *be* m *polynomials of* n *variables over* F_q*, each of which has form:*

$$P_k(x_1, \ldots, x_n) = \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_{i,j}^{(k)} x_i x_j + \sum_{i=1}^{n} \beta_i^{(k)} x_i + \gamma^{(k)},$$

whereby $\alpha_{i,j}^{(k)}, \beta_i^{(k)}, \gamma^{(k)} \in F_q$ *for all* $1 \leq k \leq m$*. Then, a MQ problem denoted by* $MQ(q, n, m)$ *is a problem of solving for indeterminates* $x_i \in F_q$ *of the random system of* m *polynomial equations* $y_k = P_k(x_1, \ldots, x_n)$ *for* $1 \leq k \leq m$*.*

Let ϕ be the *standard linear bijection* $\phi : F_{q^n} \rightarrow F_q^n$ (with some fixed basis of F_{q^n} over F_q). We introduce the important fact about this MQ problem.

Lemma 19 *(Kipnis and Shamir, 1999. [38]). Let* $F = (P_1(x_1, \ldots, x_n), \ldots, P_n(x_1, \ldots, x_n))$ *be a system of* n *multivariate polynomials of MQ Problem* $MQ(q, n, n)$ *as is in Definition 18. Then, there exists an unique univariate polynomial over* F_{q^n}*:*

$$f(x) = \sum_{i=1}^{D} a_i x^{q^{\alpha_i} + q^{\beta_i}} + \sum_{j=1}^{L} b_j x^{q^{\gamma_j}} + c,$$

where $D, L \in N$*,* $a_i, b_j, c \in F_{q^n}, \alpha_i \geq \beta_i$*,* $q^{\alpha_i} + q^{\beta_i}, q^{\gamma_j} \leq q^n - 1$ *for each* $1 \leq i \leq D, 1 \leq j \leq L$*, such that:*

$$\phi \circ f \circ \phi^{-1}(v_1, \ldots, v_n) = (P_1(v_1, \ldots, v_n), \ldots, P_n(v_1, \ldots, v_n)),$$

for $\forall (v_1, \ldots, v_n) \in F_p^n$*.*

Proof Let us fix a basis $\{\omega_1, \ldots, \omega_n\}$ of F_{q^n} as a n-dimensional vector space F_q^n over F_q. Without loss of generality we can take $\omega_1 = 1 \in F_{q^n}$. Note that an arbitrary

linearized polynomial $L(x) = \sum_{i=0}^{n-1} \alpha_i x^{q^i}$ over F_{q^n} (Definition 11) represents a linear mapping $F_q^n \to F_q^n$ which can be expressed by the associated $n \times n$ matrix B_L. There are $(q^n)^n$ distinct linearized polynomials in $F_{q^n}[x]$ while there are q^{n^2} distinct $n \times n$ matrices over F_q. Assuming that some pair of distinct linearized polynomials L_1, L_2 represents the same $n \times n$ matrix B_L yields a contradiction because the difference $(L_1 - L_2)(x)$ is the polynomial of degree at most $q^n - 1$ with q^n zeros. Therefore, there is a one-to-one correspondence between linearized polynomial in $F_{q^n}[x]$ and $n \times n$ matrices over F_q. Now, let us construct a system of n multivariate quadratic polynomials $F = (P_1(x_1, \ldots, x_n), \ldots, P_n(x_1, \ldots, x_n))$ such that:

$$P_k(x_1, \ldots, x_n) = \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_{i,j}^{(k)} x_i x_j + \sum_{i=1}^{n} \beta_i^{(k)} x_i + \gamma_i^{(k)},$$

for each $1 \leq k \leq n$. In addition to the linearity of matrix B_L, it is followed that each multivariate quadratic polynomial $P_k(x_1, \ldots, x_n)$ can be constructed by the combination of the following rudimentary operations over F_q^n.

$(x_1, \ldots, x_n) \mapsto (x_i, 0, \ldots, 0)$ This mapping is linear over F_q^n, thus we have a unique linearized polynomial $L^{(i)} \in F_q[x]$ which represents this mapping.

$(x_1, \ldots, x_n) \mapsto (x_i x_j, 0, \ldots, 0)$ This mapping is composed by multiplying the two associated linearized polynomials $L^{(i)}(x)$ and $L^{(j)}(x)$ for two mappings $(x_1, \ldots, x_n) \mapsto (x_i, 0, \ldots, 0)$ and $(x_1, \ldots, x_n) \mapsto (x_j, 0, \ldots, 0)$, respectively.

$(x_i, 0, \ldots, 0) \mapsto (0, \ldots, x_i, 0, \ldots, 0)$ This mapping is built by multiplying ω_i to the linearized polynomial $L^{(i)}$ representing the mapping $(x_1, \ldots, x_n) \mapsto (x_i, 0, \ldots, 0)$.

With these rudimentary procedures, we can generate a univariate polynomial in F_{q^n} which represents:

$$(x_1, \ldots, x_n) \mapsto (0, \ldots, P_k(x_1, \ldots, x_n), 0, \ldots, 0)$$

for each k of $1 \leq k \leq n$, and we can sum up them into a single polynomial. As we have the mapping:

$$(x_1, \ldots, x_n) \mapsto (x_i x_j, 0, \ldots, 0),$$

only the products of at most two linearized polynomials should appear in the final outcome from these rudimentary operations. Therefore, we obtained the desired result. □

From the constructive proof above the following is quite obvious.

Corollary 20 *If $F = (P_1(x_1, \ldots, x_n), \ldots, P_n(x_1, \ldots, x_n))$ a system of n multivariate polynomials of total degree at most 2 over F_p is sparse, then the corresponding univariate polynomial obtained from the procedures in Lemma 19 is also sparse.*

Now, we will define the two types of polynomials which play crucial roles throughout this paper.

Definition 21 *(Central Polynomial). Given a system of n multivariate polynomials $(P_1(x_1, \ldots, x_n), \ldots, P_n(x_1, \ldots, x_n))$ of MQ problem $MQ(q, n, n)$. A polynomial in $F_{q^n}[x]$ of the form:*

$$f(x) = \sum_{i=1}^{D} a_i x^{q^{\alpha_i} + q^{\beta_i}} + \sum_{j=1}^{L} b_j x^{q^{\gamma_j}} + c, \tag{2.1}$$

where $D, L \in N$, $a_i, b_i, c \in F_{q^n}, \alpha_i \geq \beta_i$, $q^{\alpha_i} + q^{\beta_i}, q^{\gamma_j} \leq q^n - 1$ for each $1 \leq i \leq D, 1 \leq j \leq L$, is called a central polynomial of multivariate quadratic cryptosystem based on $MQ(q, n, n)$.

The term "central" purely comes from a cryptographic reason for the design methods of trapdoor structure commonly built in the concrete multivariate quadratic cryptosystem, whereby the central polynomial of Definition 21 appears at the center of the composition of 3 secret mappings over $F_{q^n} \cong F_q^n$. It should be emphasized that

we also apply Kipnis and Shamir's lemma Lemma 19 to express each mapping of $MQ(q, n, n)$ itself by this central polynomial.

When a central polynomial has no linearized and constant terms, the polynomial has a special name in the following.

Definition 22 *(Dembowski-Ostrom Polynomial. [30, 31, 32, 33]). A polynomial in* $F_{q^n}[x]$ *of the form:*

$$f(x) = \sum_{i=1}^{D} a_i x^{q^{\alpha_i}+q^{\beta_i}},$$

where $D \in N$, $a_i \in F_{q^n}, \alpha_i \geq \beta_i$, $q^{\alpha_i} + q^{\beta_i} \leq q^n - 1$ *for each* $1 \leq i \leq D$, *is called a Dembowski-Ostrom polynomial.*

This quadratic *multinomial* is the source of the computational hardness gained in the MQ problem. We note that in the above definition a Dembowski-Ostrom polynomial can be expressed by a product of two linearized polynomials (Definition 11) and from Kipnis-Shamir's Lemma 19 it corresponds to the homogeneous system over F_p.

2. Hidden Field Equation Cryptosystem

Let q be some power of a prime number. *Hidden Field Equation System (HFE)* uses two finite fields F_q as a ground field and F_{q^n} as an extension field. The following trapdoor structure is designed in the MQ problem $MQ(q, n, n)$ (Definition 18) of a system of n multivariate quadratic polynomials in n indeterminates over F_q.

Definition 23 *(HFE Trapdoor Oneway Function. [5]). Let* F_q *and* F_{q^n} *be the finite fields. A HFE trapdoor oneway function* $F : F_p^n \rightarrow F_p^n$, *denote by* $HFE(q, n, n)$*-trapdoor, is defined as a system of* n *multivariate quadratic polynomials* $F = (P_1(x_1, \ldots, x_n), \ldots, P_n(x_1, \ldots, x_n))$ *over* F_q *such that:*

$$P_k(x_1, \ldots, x_n) = \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_{i,j}^{(k)} x_i x_j + \sum_{i=1}^{n} \beta_i^{(k)} x_i + \gamma_i^{(k)},$$

where $\alpha_{i,j}^{(k)}, \beta_i^{(k)}, \gamma_i^{(k)} \in F_q$ *for all* $1 \leq k \leq n$, *such that* F *is composed by a secret central polynomial* $f(x)$ *in* $F_{q^n}[x]$ *(Definition 21) and two affine bijections* L_1 *and* L_2 *over vector space* F_p^n*:*

$$F = L_1 \circ \phi \circ f \circ \phi^{-1} \circ L_2.$$

The mapping F *is a public key and the triple* (L_1, L_2, f) *is the private key (trapdoor information).*

In $HFE(q, n, n)$-trapdoor, the public-key computation is performed in such a way that for a given vector $x = (x_1, \ldots, x_n)$ in F_q^n:

$$F : F_q^n \ni x = (x_1, \ldots, x_n) \longmapsto F(x) = (P_1(x_1, \ldots, x_n), \ldots, P_n(x_1, \ldots, x_n)) \in F_q^n.$$

I.e., the public-key computation is a simple substitution of the value $x = (x_1, \ldots, x_n)$ $\in F_q^n$ into the system of n multivariate quadratic polynomials $F(x) = (P_1(x), \ldots,$ $P_n(x))$. Since all polynomial substitutions here are performed in the ground field F_q, the computation is very simple and efficient.

The private-key computation is possible in the reverse order of the public-key computation only with the knowledge of the private key (L_1, L_2, f) (i.e., trapdoor information). More specifically, for an arbitrarily given vector $z = (z_1, \ldots, z_n)$ in F_q^n, we first apply the inverse affine mapping L_1^{-1} to z to obtain $z' = (z'_1, \ldots, z'_n) \in F_q^n$:

$$L_1^{-1} : F_q^n \ni z = (z_1, \ldots, z_n) \longmapsto z' = (L_1^{-1}(z_1), \ldots, L_n^{-1}(z_n)) \in F_q^n.$$

Then we apply the inverse standard linear mapping $\phi^{-1} : F_q^n \to F_{q^n}$ to obtain the corresponding $z'' = \phi^{-1}(z'_1, \ldots, z'_n) \in F_{q^n}$:

$$\phi^{-1} : F_q^n \ni z' = (z'_1, \ldots, z'_n) \longmapsto z'' \in F_{q^n}.$$

At the center of the private-key computation on the $z'' \in F_{q^n}$ we need to solve the central polynomial equation:

$$z'' = f(x'')$$

in order to obtain the solution $x'' \in F_{q^n}$. We note that when we perform this private-key computation in HFE-trapdoor, it is always assumed that $f(x'') = z''$ has a solution. In other words, if HFE-trapdoor is used for encryption scheme, the private-key computation is for the *decryption* of some readily *encrypted* message, thus, this step is clearly solvable. If HFE-trapdoor is used for digital signature scheme, the private-key computation is for signing of some (hashed message). In this case, some redundancy bits are usually concatenated into the message bits so that we continue to perform this central step by applying different redundancy bits until the equation $z'' = f(x'')$ is solvable. Since a central polynomial $f(x)$ is not usually a permutation polynomial, these redundancy bits are used in both encryption and decryption providing an error correction effect so that it is always possible to pick up the *unique* right solution x'' in this step.

Finally, we apply the standard linear mapping $\phi : F_{q^n} \rightarrow F_q^n$ to obtain the corresponding $x' = (x'_1, \ldots, x'_n) = \phi(x'') \in F_q^n$ as:

$$\phi : F_{q^n} \ni x'' \longmapsto x' = (x'_1, \ldots, x'_n) \in F_q^n,$$

and subsequently apply the inverse affine mapping L_2^{-1} to x' to obtain the final outcome $x = (x_1, \ldots, x_n) \in F_q^n$:

$$L_2^{-1} : F_q^n \ni x' = (x'_1, \ldots, x'_n) \longmapsto (x_1, \ldots, x_n) \in F_q^n.$$

It should be reminded that the efficiency of central step

$$z'' = f(x'')$$

depends on the degree of $f(x)$ in HFE-trapdoor structure [5]. Therefore, we usually set some upper bound of the possible degree of the central polynomial equation $z'' = f(x'')$ in the key generation algorithm of the HFE system. The univariate polynomial solving of Berlekamp algorithm is often applied to $z'' = f(x'')$ to obtain the solution x'' in F_{q^n} (cf. Chapter 4 [19]).

3. Enhanced TTS Multivariate Signature Scheme

Let $q = p^n$ with p prime. The *Enhanced TTS Multivariate Signature Scheme (enTTS)* uses the ground finite field F_q. The following trapdoor structure is designed in the MQ problem $MQ(q, n, m)$ of a system of m multivariate quadratic polynomials in n indeterminates over F_q.

Definition 24 *(Enhanced TTS Trapdoor Oneway Function. [11]). Fix $n = 28$ and $m = 20$. Let q be $2^7 = 256$ and $F_q = F_{256}$ a finite field. TTS trapdoor oneway function $F : F_{256}^{28} \to F_{256}^{20}$, denote by $enTTS(256, 28, 20)$-trapdoor, is defined as a system of 20 multivariate quadratic polynomials $F = (P_8(x_1, \ldots, x_{28}), \ldots, P_{27}(x_1, \ldots, x_{28}))$ over F_{256} such that:*

$$P_k(x_0, \ldots, x_{27}) = \sum_{i=0}^{27} \sum_{j=0}^{27} \alpha_{i,j}^{(k)} x_i x_j + \sum_{i=0}^{27} \beta_i^{(k)} x_i + \gamma_i^{(k)},$$

where $\alpha_{i,j}^{(k)}, \beta_i^{(k)}, \gamma_i^{(k)} \in F_q$ for all $8 \le k \le 27$, is a public key of the TTS cryptosystem (Note: we use the index sets $\{0, \ldots, n-1\}$ and $\{0, \ldots, m-1\}$). $F : F_{256}^{28} \to F_{256}^{20}$ is composed by a secret multivariate central polynomial $f(x') = (f_8(x'_0, \ldots, x'_{27}), \ldots, f_{27}$

$(x'_0, \ldots, x'_{27}))$ *of* 28 *variables over* F_{256}*:*

$$
\begin{aligned}
f_i(x') &= x'_i + \sum_{j=1}^{7} \rho_j^{(i)} x'_j x'_{8+(i+j \mod 9)}, \text{ for } 8 \le i \le 16, \\
f_{17}(x') &= x'_{17} + \rho_1^{(17)} x'_1 x'_6 + \rho_2^{(17)} x'_2 x'_5 + \rho_3^{(17)} x'_3 x'_4 + \rho_4^{(17)} x'_9 x'_{16} \\
&\quad + \rho_5^{(17)} x'_{10} x'_{15} + \rho_6^{(17)} x'_{11} x'_{14} + \rho_7^{(17)} x'_{12} x'_{13}, \\
f_{18}(x') &= x'_{18} + \rho_1^{(18)} x'_2 x'_7 + \rho_2^{(18)} x'_3 x'_6 + \rho_3^{(18)} x'_4 x'_5 + \rho_4^{(18)} x'_{10} x'_{17} \\
&\quad + \rho_5^{(18)} x'_{11} x'_{16} + \rho_6^{(18)} x'_{12} x'_{15} + \rho_7^{(18)} x'_{13} x'_{14}, \\
f_i(x') &= x'_i + \rho_0^{(i)} x'_{i-11} x'_{i-9} + \sum_{j=19}^{i-1} \rho_{j-18}^{(i)} x'_{2(i-j)-(i \mod 2)} x'_j \\
&\quad + \rho_{i-18}^{(i)} x'_0 x'_i + \sum_{j=i+1}^{27} \rho_{j-18}^{(i)} x'_{i-j+19} x'_j \text{ for } 19 \le i \le 27,
\end{aligned}
$$

whereby all the coefficients $\rho_j^{(i)}$ *are the elements in* F_{256}*, and two affine bijections* L_1 *over* F_{256}^{20} *and* L_2 *over* F_{256}^{28} *such that:*

$$F = L_1 \circ f \circ L_2.$$

The mapping F *is the public key and the triple* $(L_1, L_2, f = (f_8, \ldots, f_{27}))$ *is the private key (trapdoor information). (We note that this trapdoor is denoted by* $TTS5$ *or Enhanced* $TTS(20, 28)$ *in the original paper (Section 11. [11].))*

In $enTTS(256, 28, 20)$-trapdoor, the public-key computation is performed in such a way that for a given vector $x = (x_0, \ldots, x_{27})$ in F_q^{28}:

$$F : F_{256}^{28} \ni x = (x_0, \ldots, x_{27}) \longmapsto (P_8(x_0, \ldots, x_{27}), \ldots, P_{27}(x_0, \ldots, x_{27})) \in F_{256}^{20}.$$

I.e., the public-key computation is a simple substitution of the value $x = (x_0, \ldots, x_{27})$ into the system of 20 multivariate quadratic polynomials $F(x) = (P_8(x), \ldots, P_{27}(x))$.

The private-key computation is possible in the reverse order of the public-key computation only with the knowledge of the private key $(L_1, L_2, f = (f_8, \ldots, f_{27}))$ (i.e., trapdoor information). More specifically, for an arbitrarily given vector $z = (z_8, \ldots, z_{27})$ in F_q^{28}, we first apply the inverse affine mapping L_1^{-1} to z to obtain $z' = (z'_8, \ldots, z'_{27}) \in F_q^{20}$:

$$L_1^{-1} : F_{256}^{20} \ni z = (z_8, \ldots, z_{27}) \longmapsto z' = (L_1^{-1}(z), \ldots, L_{20}^{-1}(z)) \in F_{256}^{20}.$$

At the center of this private-key inversion, with the $z' = (z'_8, \ldots, z'_{27}) \in F_{256}^{20}$ we solve the central polynomial equation:

$$(z'_8, \ldots, z'_{27}) = (f_8(x'_0, \ldots, x'_{27}), \ldots, f_{27}(x'_0, \ldots, x'_{27})),$$

in order to obtain the solution $x' = (x'_0, \ldots, x'_{27}) \in F_{256}^{28}$.

First we assign 7 random elements in F_{256} to 7 variables $x'_1, \ldots, x'_7$ in the system $f(x') = (f_8(x'), \ldots f_{27}(x'))$. Observe that this substitution turns the system $f(x')$ into a linear system $f\mid_{(x'_1,\ldots,x'_7)} (x'_8, \ldots, x'_{27})$ of equations in 20 indeterminates $x'_8, \ldots, x'_{16}$ in F_{256}. Therefore, we can apply Gaussian Elimination in order to solve $x'_8, \ldots, x'_{27}$ if possible. Otherwise we assign another 7 random elements in F_{256} to 7 variables $x'_1, \ldots, x'_7$ and repeat the process.

For x'_{17}, x'_{18}, we serially obtain their value by substituting the previous values of x''s into f_{18}, f_{18} and solve them. Finally, a random value for x'_0 is assigned and solve the last systems for $x'_{19}, \ldots, x'_{27}$. At most 9 possible values do not allow a solution for $x'_{19}, \ldots, x'_{27} \in F_{256}$. If we cannot solve at once, we repeat this random assignments.

Finally, we apply the inverse affine mapping L_2^{-1} to $x' = (x'_0, \ldots, x'_{27})$ to obtain the final outcome $x = (x_0, \ldots, x_{27}) \in F_{256}^{28}$:

$$L_2^{-1} : F_{256}^{28} \ni x' = (x'_0, \ldots, x'_{27}) \longmapsto x = (x_0, \ldots, x_{27}) \in F_{256}^{28}.$$

In contrast to HFE-trapdoor, the enTTS-trapdoor does not perform any arithmetic operation in the extension fields of F_{256}. Therefore, one may wonder if it is valid to introduce the notions of univariate central polynomials and Dembowski-Ostrom polynomials in this MQ problem. However, we introduce an appropriate generalization of both HFE-trapdoor of *two-field* type and enTTS-trapdoor of *one-field* type multivariate quadratic cryptosystems in the next subsection.

4. On the General Multivariate Quadratic Cryptosystem

In order to deal with the general MQ problem and its Weil sum application, we limit the scope of the multivariate quadratic cryptosystems in this paper by taking the following two assumptions.

1. We let p be a prime (2 or odd) and $q = p^n$ a n-th power of p. The two finite fields F_p and its extension F_q are the primary mathematical structures for the construction of our generic multivariate quadratic cryptosystem and the trapdoor structures in the system. As is often the case, the case $p = 2$ is important for many practical systems.

2. When a given MQ problem $MQ(p, n, m)$ has $m < n$ with the system of $F(x) = (P_1(x_1, \ldots, x_n), \ldots, P_m(x_1, \ldots, x_n))$: m polynomials in n indeterminates over F_p, the system F cannot induce a mapping $F_p^n \mapsto F_p^n$ by itself so that the connection with the univariate representation in Kipnis-Shamir's Lemma 19 is lost. Therefore, in order to recover this connection, we embed the original system $F = (P_1, \ldots, P_m)$ into the extended system $\overline{F} = (P_1, \ldots, P_m, P_{m+1}, \ldots, P_n)$ such that, for example:

$$P_{m+1}(x_1, \ldots, x_n) = \cdots = P_n(x_1, \ldots, x_n) = 0.$$

In this way, we can consider the extended MQ problem $MQ(p, n, n)$ whose system has the corresponding univariate central polynomial by Lemma 19. It is easy to see that any solution found for $MQ(p, n, n)$ is also a solution of the embedded $MQ(p, n, m)$.

Our main goal in this paper is to identify and analyze some *nontrivial* classes of the weak polynomial structures that exist in the generic multivariate quadratic cryptosystem based on MQ problem $MQ(p, n, n)$. We denote by $MQ(p, n, n)$-*trapdoor* an arbitrary trapdoor structure designed into the public system $F(x) = (P_1(x_1, \ldots, x_n), \ldots, P_n(x_1, \ldots, x_n))$: n multivariate quadratic polynomials in n indeterminates over F_p whereby the equation $z = F(x)$ for given $z \in F_p^n$ is secretly inverted with the designed trapdoor information. Therefore, regardless of the concrete types of trapdoor expressed by $MQ(p, n, n)$-*trapdoor*, we can always work on the corresponding univariate polynomial over the extension field F_q of form:

$$f(x) = \sum_{i=1}^{D} a_i x^{p^{\alpha_i}+p^{\beta_i}} + \sum_{j=1}^{L} b_j x^{p^{\gamma_j}} + c,$$

which is identical to that of central polynomial in Definition 21 for the system $F = (P_1(x_1, \ldots, x_n), \ldots, P_n(x_1, \ldots, x_n))$ over F_p. Similarly, we may have Dembowski-Ostrom polynomial:

$$f(x) = \sum_{i=1}^{D} a_i x^{p^{\alpha_i}+p^{\beta_i}},$$

when each multivariate polynomial P_k of the system is a quadratic form (i.e. homogeneous quadratic polynomial).

For $m < n$, the system does not naturally induce the univariate polynomial representation so the connection to the univariate polynomial is lost. In order to regain the connection of the systems $MQ(2, n, m)$-trapdoor with $m < n$ to the univariate representations, we simply assume that $MQ(2, n, m)$-trapdoor is *embedded* into

$MQ(2, n, n)$-trapdoor so that the solutions found on $MQ(2, n, n)$-trapdoor are also the solutions for this embedded $MQ(2, n, m)$-trapdoor.

It should also be noted that the enTTS-trapdoor in Definition 24 has no Dembowski-Ostrom polynomial in its key generation, and thus its trapdoor function is not homogeneous. And also the inversion of the multivariate central polynomial mappings:

$$f(x') = (f_8(x'_1, \ldots, x'_n), \ldots, f_{28}(x'_1, \ldots, x'_n))$$

over F_{256} are performed without using bigger finite field. In this case we apply the second assumption to expand the system given by enTTS-trapdoor so that the extended MQ problem can supersede the original system given by the enTTS-trapdoor.

In the key generation algorithm of the enTTS-trapdoor, there is no restriction of upper bound to be set on the central polynomials. Therefore, in the following we do not set the upper bound of central polynomials used in the generic MQ problems.

CHAPTER III

WEIL SUM EVALUATION OF CENTRAL POLYNOMIALS

A. Introduction

Let F_q be a finite field of characteristic p (2 or any odd prime) and order $q = p^n$, where n is the extension degree of F_q over F_p. In this chapter we fix a MQ problem $MQ(p, n, n)$ (Definition 18) and assume that some generic trapdoor structure $MQ(p, n, n)$-trapdoor is defined. We introduce the explicit Weil sum evaluation schemes of the central polynomials that express the generic MQ problems.

In Section B we introduce a Weil sum of central polynomials in Definition 21 and show that we can simplify the form of the central polynomials to facilitate Weil sum evaluation. In Section C, we define a certain linearized polynomial for the simplified central polynomials in the previous section. It is shown that this linearized polynomial governs the computational efficiency as well as the final Weil sum value of the central polynomial. Section D introduces a new Weil sum evaluation algorithm of the generic central polynomials for the MQ problem. This algorithm is able to compute the absolute value of the Weil sum of central polynomial in time empirically efficient in terms of sparsity of the input central polynomial, the dimension of the auxiliary linearized polynomial in Section C and the extension degree n.

B. Simplification of Central Polynomial

Let F_q be a finite field of characteristic p (2 or any odd prime) and order $q = p^n$. n is the extension degree of F_q over F_p. Denote by $S(a_1, \ldots, a_D, b_1, \ldots, b_L, c)$ (or simply S) the Weil sum (Definition 15) of a central polynomial $f(x) = \sum_{i=1}^{D} a_i x^{p^{\alpha_i}+p^{\beta_i}} + \sum_{j=1}^{L} b_j x^{p^{\gamma_j}} + c \in F_q[x]$ (Definition 21). Explicitly, with canonical additive character

χ_1 of F_q, we consider:

$$S = S(a_1, \ldots, a_D, b_1, \ldots, b_L, c) = \sum_{x \in F_q} \chi_1(\sum_{i=1}^{D} a_i x^{p^{\alpha_i}+p^{\beta_i}} + \sum_{j=1}^{L} b_j x^{p^{\gamma_j}} + c).$$

Applying the property of additive character in Theorem 13 to the *constant* term c of $f(x)$ in the Weil sum S yields an equivalent Weil sum:

$$S = \chi_1(c)\{\sum_{x \in F_q} \chi_1(\sum_{i=1}^{D} a_i x^{p^{\alpha_i}+p^{\beta_i}} + \sum_{j=1}^{L} b_j x^{p^{\gamma_j}})\}.$$

That is, one can separately treat the image of c by character χ_1 when we evaluate the Weil sum value S. Therefore, without loss of generality, we can always assume that the constant term of central polynomials be zero ($f(0) = c = 0$) and may separately deal with the value $\chi_1(c) = \chi_1(f(0))$ at the final step in Weil sum evaluation algorithm.

In the following we will show that we can also simplify the *linearized* terms $\sum_{j=1}^{L} b_j x^{p^{\gamma_j}}$ in the central polynomial $f(x) = \sum_{i=1}^{D} a_i x^{p^{\alpha_i}+p^{\beta_i}} + \sum_{j=1}^{L} b_j x^{p^{\gamma_j}}$. The newly introduced coefficients $A_i \in F_q$ and parameters $t_i, y_i, s_i \in Z$ and $b \in F_q$ in the following theorem will be later justified in the specification of the subsequent Theorem 26 regarding the auxiliary linearized polynomial of central polynomials.

Theorem 25 *(Simplification of Central Polynomials). Let $f(x)$ be a central polynomial over F_q of Definition 21 (with $f(0) = 0$). Assume that we set the new coefficients A_i such that $A_i^{p^{t_i}} = a_i \in F_q$ $(1 \le i \le D)$ and parameters $t_i, y_i, s_i \in Z$, and $b \in F_q$ such that $t_i \equiv \beta_i - \beta_1 \mod n$ $(1 \le i \le D)$, and $y_i = n - s_i$ $(2 \le i \le D)$, $s_i = \alpha_i - \beta_i \ge 0$ $(1 \le i \le D)$ and $b = \sum_{j=1}^{L} b_j^{p^{e-\gamma_j}}$. Then, we can express the Weil sum $S = S(a_1, \ldots, a_D, b_1, \ldots, b_L)$ of $f(x)$ as:*

$$S = \sum_{x \in F_q} \chi_1(\sum_{i=1}^{D} A_i x^{p^{s_i}+1} + b^{p^{\beta_1}} x).$$

We call the polynomial $\sum_{i=1}^{D} A_i x^{p^{s_i}+1} + b^{p^{\beta_1}} x$ the simplified central polynomial of $f(x)$.

Proof The proof of the simplification consists of two parts. First, we simplify the linearized terms $\sum_{j=1}^{L} b_j x^{p^{\gamma_j}}$ in central polynomial $f(x)$ into a *single linear* term by using Theorem 13. In the transformations below, we repeatedly apply Theorem 13 when splitting and joining the arguments of χ_1 and taking the various powers of p inside each argument. The first transformation is the following.

$$\begin{aligned}
\sum_{x\in F_q} \chi_1(\sum_{i=1}^{D} a_i x^{p^{\alpha_i}+p^{\beta_i}} + \sum_{j=1}^{L} b_j x^{p^{\gamma_j}}) &= \sum_{x\in F_q} \chi_1(\sum_{i=1}^{D} a_i x^{p^{\alpha_i}+p^{\beta_i}})\chi_1(\sum_{j=1}^{L} b_j x^{p^{\gamma_j}}) \\
&= \sum_{x\in F_q} \chi_1(\sum_{i=1}^{D} a_i x^{p^{\alpha_i}+p^{\beta_i}}) \prod_{j=1}^{L} \chi_1(b_j x^{p^{\gamma_j}}) \\
&= \sum_{x\in F_q} \chi_1(\sum_{i=1}^{D} a_i x^{p^{\alpha_i}+p^{\beta_i}}) \prod_{j=1}^{L} \chi_1(b_j^{p^{e-\gamma_j}} x^{p^e}) \\
&= \sum_{x\in F_q} \chi_1(\sum_{i=1}^{D} a_i x^{p^{\alpha_i}+p^{\beta_i}}) \prod_{j=1}^{L} \chi_1(b_j^{p^{e-\gamma_j}} x) \\
&= \sum_{x\in F_q} \chi_1(\sum_{i=1}^{D} a_i x^{p^{\alpha_i}+p^{\beta_i}})\chi_1(\sum_{j=1}^{L} b_j^{p^{e-\gamma_j}} x) \\
&= \sum_{x\in F_q} \chi_1(\sum_{i=1}^{D} a_i x^{p^{\alpha_i}+p^{\beta_i}})\chi_1(bx) \\
&= \sum_{x\in F_q} \chi_1(\sum_{i=1}^{D} a_i x^{p^{\alpha_i}+p^{\beta_i}} + bx).
\end{aligned}$$

Therefore, the linearized terms $\sum_{j=1}^{L} b_j x^{p^{\gamma_j}}$ of $f(x)$ is turned into a single linear term bx where $b = \sum_{j=1}^{L} b_j^{p^{e-\gamma_j}}$ and thus we have $S = S(a_1, \ldots, a_D, b_1, \ldots, b_L) = S(a_1, \ldots, a_D, b)$.

Next, we replace the coefficients a_i's with the new coefficients $A_i \in F_q$ under the new integer parameters $t_i \equiv \beta_i - \beta_1 \mod n$ $(1 \le i \le D)$, $y_i = n - s_i$ $(2 \le i \le D)$,

$s_i = \alpha_i - \beta_i \geq 0$ $(1 \leq i \leq D)$. We have:

$$\begin{aligned}
S(a_1, \ldots, a_D, b) &= \sum_{x \in F_q} \chi_1(\sum_{i=1}^{D} a_i x^{p^{\alpha_i}+p^{\beta_i}} + bx) \\
&= \sum_{x \in F_q} \prod_{i=1}^{D} \chi_1(a_i x^{p^{\alpha_i}+p^{\beta_i}}) \chi_1(bx) \\
&= \sum_{x \in F_q} \prod_{i=1}^{D} \chi_1(a_i x^{p^{\beta_i}(p^{s_i}+1)}) \chi_1(bx) \\
&= \sum_{x \in F_q} \prod_{i=1}^{D} \chi_1(A_i^{p^{t_i}} (x^{p^{\beta_1}})^{p^{t_i}(p^{s_i}+1)}) \chi_1(b^{p^{\beta_1}} x^{p^{\beta_1}}) \\
&= \sum_{u \in F_q} \prod_{i=1}^{D} \chi_1((A_i u^{p^{s_i}+1})^{p^{t_i}}) \chi_1(b^{p^{\beta_1}} u) \text{ [Note: } u = x^{p^{\beta_1}}] \\
&= \sum_{u \in F_q} \chi_1(\sum_{i=1}^{D} A_i u^{p^{s_i}+1} + b^{p^{\beta_1}} u).
\end{aligned}$$

Therefore, we obtain the simplification $S(a_1, \ldots, a_D, b) = S(A_1, \ldots, A_D, b)$, i.e., the equivalent Weil sum of the form:

$$S = \sum_{x \in F_q} \chi_1(\sum_{i=1}^{D} A_i x^{p^{s_i}+1} + b^{p^{\beta_1}} x),$$

with the simplified central polynomial $\sum_{i=1}^{D} A_i x^{p^{s_i}+1} + b^{p^{\beta_1}} x$. We obtained the desired result. □

This theorem says that at the level of Weil sum values the Weil sum of generic central polynomial is equal to that of special type of central polynomials of form:

$$\sum_{i=1}^{D} A_i x^{p^{s_i}+1} + b^{p^{\beta_1}} x.$$

This is the reason we name this type a simplified central polynomial.

C. Weil Sum of Central Polynomials

It is shown in Theorem 25 the Weil sum of a central polynomial $\sum_{i=1}^{D} a_i x^{p^{\alpha_i}+p^{\beta_i}} + \sum_{j=1}^{L} b_j x^{p^{\gamma_j}}$ (no constant term) over F_q is equivalent to that of the corresponding simplified central polynomial of the form $\sum_{i=1}^{D} A_i x^{p^{s_i}+1} + b^{p^{\beta_1}} x$ with specially introduced new coefficients $A_i \in F_q$ and parameters $t_i, y_i, s_i \in Z$ and $b \in F_q$. In the following theorem, we will justify their occurrences. We will deduce the *auxiliary linearized polynomial*, denoted by $T_D(x)$ in $F_q[x]$. This linearized polynomial naturally appears during the calculation in the proof, and afterward enables us to compute the concrete absolute value of the Weil sum. As is common in the proof techniques of *explicit evaluation of Weil sum*, we start with taking the *product* of the Weil sum and its *conjugate.* Note that p can be either 2 or any odd prime.

Theorem 26 *(Auxiliary Linearized Polynomials. cf. Theorem 1.4. [33]). Let $f(x)$ be a central polynomial over F_q of Definition 21 (with $f(0) = 0$), and let S be the Weil sum of $f(x)$. Then, the product $|S|^2 = S\overline{S}$ is:*

$$|S|^2 = q \sum_{T_D(w)=0, w \in F_q} \chi_1(\sum_{i=1}^{D} A_i w^{p^{s_i}+1} + b^{p^{\beta_1}} w),$$

whereby A_i's are the new coefficients such that $A_i^{p^{t_i}} = a_i$ $(1 \le i \le D)$, and t_i, y_i, s_i and b are the new parameters such that $t_i \equiv \beta_i - \beta_1 \bmod n$ $(1 \le i \le D)$, $y_i = n - s_i$ $(2 \le i \le D)$, $s_i = \alpha_i - \beta_i \ge 0$ $(1 \le i \le D)$ and $b = \sum_{j=1}^{L} b_j^{p^{e-\gamma_j}}$. The index w of the outer sum runs throughout the set of roots in F_q of a linearized polynomial defined as:

$$T_D(w) = A_1^{p^{s_1}} w^{p^{2s_1}} + A_1 w + \sum_{i=2}^{D} [A_i^{p^{s_1}} w^{p^{s_1+s_i}} + (A_i w)^{p^{s_1+y_i}}].$$

Proof From Theorem 25 we can work on the simplified central polynomial $\sum_{i=1}^{D} A_i$

$x^{p^{s_i}+1} + b^{p^{s_1}}x$ over F_q. That is, the Weil sum is expressed by:

$$S = S(A_1, \ldots, A_D, b^{p^{\beta_1}}) = \sum_{x \in F_q} \chi_1(\sum_{i=1}^{D} A_i x^{p^{s_i}+1} + b^{p^{\beta_1}}x).$$

Now, let us take the product $|S|^2 = S(A_1, \ldots, A_D, b^{p^{\beta_1}})\overline{S(A_1, \ldots, A_D, b^{p^{\beta_1}})}$. We will show that the linearized polynomial $T_D(x)$ naturally appears during the course of simplification of this product. As is in the transformation in Theorem 25, we repeatedly apply Theorem 13 when splitting and joining the arguments of χ_1 and taking the various powers of p inside each argument. First we have:

$$\begin{aligned}
|S|^2 &= S\overline{S} \\
&= \{\sum_{u \in F_q} \chi_1(\sum_{i=1}^{D} A_i u^{p^{s_i}+1} + b^{p^{\beta_1}}u)\} \cdot \{\sum_{v \in F_q} \overline{\chi_1}(\sum_{i=1}^{D} + A_i v^{p^{s_i}+1} + b^{p^{\beta_1}}v)\} \\
&= \{\sum_{u \in F_q} \chi_1(\sum_{i=1}^{D} A_i u^{p^{s_i}+1} + b^{p^{\beta_1}}u)\} \cdot \{\sum_{v \in F_q} \chi_1(\sum_{i=1}^{D} - A_i v^{p^{s_i}+1} - b^{p^{\beta_1}}v)\} \\
&= \sum_{u,v \in F_q} \chi_1(\sum_{i=1}^{D} A_i[u^{p^{s_i}+1} - v^{p^{s_i}+1}] + b^{p^{\beta_1}}(u - v)) \\
&= \sum_{w,v \in F_q} \chi_1(\sum_{i=1}^{D} A_i[(w+v)^{p^{s_i}+1} - v^{p^{s_i}+1}] + b^{p^{\beta_1}}w) \\
&= \sum_{w,v \in F_q} \chi_1(\sum_{i=1}^{D} A_i(w^{p^{s_i}+1} + wv^{p^{s_i}+1} + vw^{p^{s_i}+1}) + b^{\beta^{s_1}}w) \\
&= \sum_{w,v \in F_q} \chi_1(\sum_{i=1}^{D} A_i w^{p^{s_i}+1} + b^{p^{\beta_1}}w) \cdot \chi_1(\sum_{i=1}^{D} A_i(wv^{p^{s_i}} + vw^{p^{s_i}})).
\end{aligned}$$

We can further simplify the character $C_{w,v} = \chi_1(\sum_{i=1}^{D} A_i(wv^{p^{s_i}} + vw^{p^{s_i}}))$ as follows.

$$\begin{aligned}
C_{w,v} &= \chi_1(\sum_{i=1}^{D} A_i(wv^{p^{s_i}} + vw^{p^{s_i}})) \\
&= \chi_1(A_1 vw^{p^{s_1}} + A_1 wv^{p^{s_1}} + \sum_{i=2}^{D} A_i vw^{p^{s_i}} + \sum_{i=2}^{D} A_i wv^{p^{s_i}}) \\
&= \chi_1(A_1^{p^{s_1}} v^{p^{s_1}} w^{p^{2s_1}} + A_1 wv^{p^{s_1}} + \sum_{i=2}^{D} A_i^{p^{s_1}} v^{p^{s_1}} w^{p^{s_i+s_1}} + \sum_{i=2}^{D} (A_i w)^{p^{s_1+y_i}} v^{p^{s_1}}) \\
&= \chi_1(v^{p^{s_1}} T_D(w)).
\end{aligned}$$

Therefore, we have:

$$|S|^2 = \sum_{w \in F_q} \chi_1(\sum_{i=1}^{D} A_i w^{p^{s_i}+1} + b^{p^{\beta_1}} w) \cdot \sum_{v \in F_q} \chi_1(v^{p^{s_1}} T_D(w)).$$

Recall that $v^{p^{s_1}}$ runs throughout F_q as v runs throughout F_q. Also by Theorem 14, the inner sum $\sum_{v \in F_q} \chi_1(v^{p^{s_1}} T_D(w))$ is zero unless $T_n(w) = 0$ for the index $w \in F_q$ of the outer sum, because otherwise $v^{p^{s_1}} T_D(w)$ also runs throughout F_q as v runs throughout F_q. Therefore we have:

$$|S|^2 = q \sum_{T_D(w)=0, w \in F_q} \chi_1(\sum_{i=1}^{D} A_i w^{p^{s_i}+1} + b^{p^{\beta_1}} w),$$

which is the desired result. □

Finally, we will show a lemma regarding the set of roots of the auxiliary linearized polynomial of the central polynomial.

Lemma 27 *(Roots of Auxiliary Linearized Polynomial. Lemma 3.4 [33]). Let $T_D(x)$ be an auxiliary linearized polynomial over F_q defined in Theorem 26. Suppose that $\varepsilon = \gcd_{2 \le i \le D}(2s_1, s_1 + s_i, s_1 + y_i, n)$. Then, the set of roots of $T_D(x)$ forms a linear subspace of F_q over F_{p^ε} and is isomorphic to $F_{p^{t\varepsilon}}$ for some integer $t \in Z$.*

Proof For any monomial x^{p^α} in $T_D(x) = A_1^{p^{s1}} x^{p^{2s_1}} + A_1 x + \sum_{i=2}^{D}[A_i^{p^{s1}} x^{p^{s_1+s_i}} + (A_i x)^{p^{s_1+y_i}}]$, the exponent α is divisible by the greatest common divisor ε. Therefore for any $u \in F_{p^\varepsilon}$, $u^{p^\alpha} = (((u^{p^\varepsilon})^{p^\varepsilon}) \cdots)^{p^\varepsilon} = u$ (α/ε times). Hence we have:

$$\begin{aligned} T_D(ux) &= A_1^{p^{s1}} (ux)^{p^{2s_1}} + A_1 ux + \sum_{i=2}^{D}[A_i^{p^{s1}} (ux)^{p^{s_1+s_i}} + (A_i ux)^{p^{s_1+y_i}}] \\ &= A_1^{p^{s1}} ux^{p^{2s_1}} + A_1 ux + \sum_{i=2}^{D}[A_i^{p^{s1}} ux^{p^{s_1+s_i}} + u(A_i x)^{p^{s_1+y_i}}] \\ &= uT_D(x), \end{aligned}$$

for all $u \in F_{p^\varepsilon}$. That is, the set of roots of $T_D(x)$ is a linear subspace of F_q over F_{p^ε}. By setting $t \in Z$ the dimension of this subvector space over F_{p^ε}, the set of the roots of $T_D(x)$ is $F_{p^\varepsilon}^t \simeq F_{p^{t\varepsilon}}$ and its cardinality is $p^{\varepsilon t}$. □

D. Weil Sum Algorithm for Central Polynomial

For finite fields of characteristic $p = 2$, the Weil sum with canonical additive character is guaranteed to be *real*. It should be noted that this fact is quite different from those in the cases when p is odd prime [33]. We have the following lemma for $p = 2$.

Lemma 28 *(Character ($p = 2$)). Let F_q be of characteristic $p = 2$. Then, for any $u \in F_q$, $\chi_1(u)$ is real.*

Proof It is a simple matter to show that from the definition of canonical additive character, for any $u \in F_q$, we have:

$$\chi_1(u) = \exp(\frac{2\pi i}{p} Tr(u)) = \exp(\pi i Tr(u)).$$

Since the image of $u \in F_q$ by the absolute trace function $Tr(u) = Tr_1(u)$ is in $\{0, 1\} = F_2$, the value $\chi_1(u)$ is either 1 or -1, which is real. □

It follows by Lemma 28 that we also have:

Corollary 29 *Let $p = 2$ and S the Weil sum of central polynomial $f(x)$ as in Theorem 26. Then, S is real and $|S|^2 = S^2$.*

This corollary readily embraces the important idea for the efficient Weil sum algorithm (for $p = 2$): for finite fields of $p = 2$, if we can compute the product of the Weil sum S and its conjugate $\overline{S}$ as is in Theorem 26, then we can obtain the absolute value of the Weil sum $|S|$. To see this more specifically, let $f(x)$ be a central polynomial

$$f(x) = \sum_{i=1}^{D} a_i x^{p^{\alpha_i}+p^{\beta_i}} + \sum_{j=1}^{L} b_j x^{p^{\gamma_j}} \in F_q[x].$$

Theorem 25 says that we can express its Weil sum $S = S(a_1, \ldots, a_D, b_1, \ldots, b_L)$ as

$$S = \sum_{x \in F_q} \chi_1(\sum_{i=1}^{D} A_i x^{p^{s_i}+1} + b^{p^{\beta_1}} x),$$

where $\sum_{i=1}^{D} A_i x^{p^{s_i}+1} + b^{p^{\beta_1}} x$ is the simplified central polynomial with coefficients $A_i^{p^{t_i}} = a_i$ $(1 \le i \le D)$ and parameters $t_i \equiv \beta_i - \beta_1 \bmod n$ $(1 \le i \le D)$, $y_i = n - s_i$ $(2 \le i \le D)$, $s_i = \alpha_i - \beta_i \ge 0$ $(1 \le i \le D)$, and $b = \sum_{j=1}^{L} b_j^{p^{e-\gamma_j}}$ as usual. By the theorem of auxiliary linearized polynomial (Theorem 26), the product $|S|^2 = S\overline{S}$ is expressed by:

$$|S|^2 = q \sum_{T_D(w)=0, w \in F_q} \chi_1(\sum_{i=1}^{D} A_i w^{p^{s_i}+1} + b^{p^{\beta_1}} w),$$

where $T_D(w) = A_1^{p^{s_1}} w^{p^{2s_1}} + A_1 w + \sum_{i=2}^{D} [A_i^{p^{s_1}} w^{p^{s_1+s_i}} + (A_i w)^{p^{s_1+y_i}}]$ is the auxiliary linearized polynomial in $F_q[x]$, and also Corollary 29 yields

$$S = \pm\sqrt{S^2} = \pm\sqrt{|S|^2}.$$

Now, let us consider some basis $\{\omega_1, \ldots, \omega_n\}$ of F_q as a vector space $F_q \cong F_p^n$. Then, T_D is actually a linear mapping over F_p^n by the same reason in Lemma 27.

More specifically, we have:

$$T_D(ux) = uT_D(x),$$

for all $u \in F_p$ and $x \in F_q$. (Obviously, the exponent $p^0 = 1$ of $u = u^{p^0}$ divides the exponents of any monomial appearing in $T_D(x)$.) Let us take an $n \times n$ matrix $B = (b_{ik}), 1 \leq i, k \leq n$ over F_p that represents the corresponding mapping T_D over F_q. In other words, we have for each ω_i in $\{\omega_1, \ldots, \omega_n\}$,

$$T_D(\omega_i) = \sum_{k=1}^{n} b_{ik}\omega_k,$$

$b_{ik} \in F_p$ and equivalently:

$$y_1\omega_1 + \cdots + y_n\omega_n = T_D(x_1\omega_1 + \cdots + x_n\omega_n) \Longleftrightarrow (y_1, \ldots, y_n) = (x_1, \ldots, x_n)B,$$

for $(x_1, \ldots, x_n), (y_1, \ldots, y_n) \in F_p^n$.

In order to actually compute the partial sum in the product $|S|^2$, we need some representation of the set of the roots of the equation $T_D(x) = 0$. Suppose $r = rank(B)$ is the rank of the matrix B. Then, there are p^{n-r} roots of $T_D(w) = 0$ in F_q (note that $w = 0$ is always a root). So let $l = n - r$ and assume that some basis $\{\eta_1, \ldots, \eta_l\} \subset F_q$ forms the set of the roots of $T_D(w) = 0$ which is a subvector space of F_p^n. Then, from the properties of linearity and powers of p of trace function in Theorem 12, we have for any $\eta = \sum_{i=1} x_i\eta_i \in ker(T_D)$ with each $x_i \in F_p$:

$$
\begin{aligned}
Tr(\sum_{i=1}^{D} A_i\eta^{p^{s_i}+1} + b^{p^{\beta_1}}\eta) &= Tr(\sum_{i=1}^{D} A_i(\sum_{j=1}^{l} x_j\eta_j)^{p^{s_i}+1} + b^{p^{\beta_1}}(\sum_{j=1}^{l} x_j\eta_j)) \\
&= Tr(\sum_{i=1}^{D} A_i(\sum_{j_1=1}^{l} x_{j_1}\eta_{j_1})\cdot(\sum_{j_2=1}^{l} x_{j_2}\eta_{j_2})^{p^{s_i}} + b^{p^{\beta_1}}(\sum_{j=1}^{l} x_j\eta_j)) \\
&= Tr(\sum_{i=1}^{D} A_i(\sum_{j_1=1}^{l} x_{j_1}\eta_{j_1})\cdot(\sum_{j_2=1}^{l} x_{j_2}^{p^{s_i}}\eta_{j_2}^{p^{s_i}}) + b^{p^{\beta_1}}(\sum_{j=1}^{l} x_j\eta_j)) \\
&= \sum_{i=1}^{D}\sum_{j_1=1}^{l}\sum_{j_2=1}^{l} Tr(A_i x_{j_1} x_{j_2}^{p^{s_i}}\eta_{j_1}\eta_{j_2}^{p^{s_i}}) + \sum_{j=1}^{l} Tr(b^{p^{\beta_1}} x_j\eta_j) \\
&= \sum_{i=1}^{D}\sum_{j_1=1}^{l}\sum_{j_2=1}^{l} x_{j_1} x_{j_2}^{p^{s_i}} Tr(A_i\eta_{j_1}\eta_{j_2}^{p^{s_i}}) + \sum_{j=1}^{l} x_j Tr(b^{p^{\beta_1}}\eta_j) \\
&= \sum_{i=1}^{D}\sum_{j_1=1}^{l}\sum_{j_2=1}^{l} x_{j_1} x_{j_2} Tr(A_i\eta_{j_1}\eta_{j_2}^{p^{s_i}}) + \sum_{j=1}^{l} x_j Tr(b^{p^{\beta_1}}\eta_j).
\end{aligned}
$$

Therefore, we have:

$$
\chi_1(\sum_{i=1}^{D} A_i\eta^{p^{s_i}+1} + b^{p^{\beta_1}}\eta) = \exp(\frac{2\pi i}{p}(\sum_{i=1}^{D}\sum_{j_1=1}^{l}\sum_{j_2=1}^{l} x_{j_1} x_{j_2} Tr(A_i\eta_{j_1}\eta_{j_2}^{p^{s_i}}) + \sum_{j=1}^{l} x_j Tr(b^{p^{\beta_1}}\eta_j))).
$$

Henceforth, let us consider the case $\mathbf{p = 2}$. Now we have:

$$
\chi_1(\sum_{i=1}^{D} A_i\eta^{2^{s_i}+1} + b^{2^{\beta_1}}\eta) = \exp(\pi i(\sum_{i=1}^{D}\sum_{j_1=1}^{l}\sum_{j_2=1}^{l} x_{j_1} x_{j_2} Tr(A_i\eta_{j_1}\eta_{j_2}^{2^{s_i}}) + \sum_{j=1}^{l} x_j Tr(b^{2^{\beta_1}}\eta_j))).
$$

By pre-computing the trace values:

$$
\begin{cases}
\gamma_{i,j_1,j_2} = Tr(A_i\eta_{j_1}\eta_{j_2}^{2^{s_i}}), \\
\rho_j = Tr(b^{2^{\beta_1}}\eta_j),
\end{cases}
$$

for $1 \le i \le D, 1 \le j_1, j_2 \le l$, we can evaluate the *parity* $C_{(x_1,\ldots,x_l)}$:

$$
C_{(x_1,\ldots,x_l)} = \sum_{i=1}^{D}\sum_{j_1=1}^{l}\sum_{j_2=1}^{l} x_{j_1} x_{j_2}\gamma_{i,j_1,j_2} + \sum_{j=1}^{l} x_j\rho_j \in F_2 = \{0, 1\}
$$

for each $(x_1, \ldots, x_l) \in F_2^l$. Therefore, for the image of χ_1 on the argument with $\eta \in \ker(T_D)$, we have either:

$$\chi_1(\sum_{i=1}^{D} A_i \eta^{2^{s_i}+1} + b^{2^{\beta_1}}\eta) = \exp(\pi i C_{(x_1,\ldots,x_l)}) = 1,$$

if $C_{(x_1,\ldots,x_l)} = 0$, or:

$$\chi_1(\sum_{i=1}^{D} A_i \eta^{2^{s_i}+1} + b^{2^{\beta_1}}\eta) = \exp(\pi i C_{(x_1,\ldots,x_l)}) = -1,$$

if $C_{(x_1,\ldots,x_l)} = 1$. We can combine these two cases as:

$$\chi_1(\sum_{i=1}^{D} A_i \eta^{2^{s_i}+1} + b^{2^{\beta_1}}\eta) = \exp(\pi i C_{(x_1,\ldots,x_l)}) = 1 - 2C_{(x_1,\ldots,x_l)},$$

for each root $\eta = \sum_{i=1}^{l} x_i \eta_i \in ker(T_D)$. As a result, we obtain:

$$\begin{aligned}
|S|^2 &= |S(A_1, \ldots, A_D, b^{2^{\beta_1}})|^2 \\
&= 2^n \sum_{T_D(w)=0, w\in F_q} \chi_1(\sum_{i=1}^{D} A_i w^{2^{s_i}+1} + b^{2^{\beta_1}} w) \\
&= 2^n \sum_{(x_1,\ldots,x_l)\in F_2^l, \eta=\sum_{i=1}^{l} x_i\eta_i} \chi_1(\sum_{i=1}^{D} A_i \eta^{2^{s_i}+1} + b^{2^{\beta_1}}\eta) \\
&= 2^n \sum_{(x_1,\ldots,x_l)\in F_2^l} (1 - 2C_{(x_1,\ldots,x_l)}) \\
&= 2^n (2^l - 2 \sum_{(x_1,\ldots,x_l)\in F_2^l} C_{(x_1,\ldots,x_l)}).
\end{aligned}$$

Note that since $p = 2$, the simple parity check of the value $C_{(x_1,\ldots,x_l)}$ is sufficient for obtaining the value $\chi_1(\sum_{i=1}^{D} A_i \eta^{2^{s_i}+1} + b^{2^{\beta_1}}\eta) = \exp(\pi i C_{(x_1,\ldots,x_l)})$, rather than performing complex number calculation with $\exp(\frac{2\pi}{p} i Tr_p(X))$ as for the case where p odd prime. We can formulate the above Weil sum algorithm in the following.

Algorithm 30 *(Weil Sum Algorithm ($p = 2$)). Assume that a basis $\{\omega_1, \ldots, \omega_n\}$ of $F_{2^n} \cong F_2^n$ is available before computation.*

INPUT $f(x) = \sum_{i=1}^{D} a_i x^{2^{\alpha_i}+2^{\beta_i}} + \sum_i^L b_i x^{2^{\gamma_i}}$*: a central polynomial in* $F_{2^n}[x]$.

OUTPUT $|S|$*: the absolute value of Weil sum* S *of* $f(x)$.

1. *Compute the associated auxiliary linearized polynomial* $T_D(x) \in F_{2^n}[x]$ *as in Theorem 26 (Suppose the rank of the kernel is* l*).*
2. *Compute the basis* $\{\eta_1, \ldots, \eta_l\}$ *of* $ker(T_D)$.
3. *Let* U *be* $0 \in Z$.
4. *Compute* $\gamma_{i,j_1,j_2} = Tr(A_i \eta_{j_1} \eta_{j_2}^{2^{s_i}})$ *for* $1 \le i \le D, 1 \le j_1, j_2 \le l$
5. *Compute* $\rho_j = Tr(b^{2^{\beta_1}} \eta_j)$ *for* $1 \le j \le l$.
6. *For each* $(x_1, \ldots, x_l) \in F_2^l$*, evaluate:*

$$C_{(x_1,\ldots,x_l)} = \sum_{i=1}^{D} \sum_{j_1=1}^{l} \sum_{j_2=1}^{l} x_{j_1} x_{j_2} \gamma_{i,j_1,j_2} + \sum_{j=1}^{l} x_j \rho_j \in F_2$$

   and set $U = U + C_{(x_1,\ldots,x_l)}$*. (Note: integer addition.)*
7. *Return* $2^{n/2}\sqrt{2^l - 2U}$.

Theorem 31 *(Validity and Complexity). The Weil sum algorithm in Algorithm 30 computes the absolute value* $|S|$ *of Weil sum* S *of the input central polynomial* $f(x) = \sum_{i=1}^{D} a_i x^{2^{\alpha_i}+2^{\beta_i}} + \sum_i^L b_i x^{2^{\gamma_i}}$ *in* $F_{2^n}[x]$ *in time:*

$$O(C_{DL} l^2 (n^3 + 2^l)),$$

where l *is the dimension of the kernel of the auxiliary linearized polynomial* $T_D(x)$ *and* $C_{DL} = D + L$ *is the sparsity of* $f(x)$.

Proof Suppose that the basic arithmetic operations for the elements in F_q costs $O(\log^2 q)$ time in RAM. When $q = 2^n$, it takes $O(n^2)$ time. The estimate of each step of the algorithm is:

1. A_i is obtained from 2^{n-t_i}-th power of a_i, thus in $O(C_{DL}n^3)$ time.
2. Performing Gaussian Elimination on B to obtain the basis $\{\eta_1, \ldots, \eta_l\}$ takes $O(n^3)$ time.
3. $O(1)$ time.
4. Trace $Tr(x) = x + x^2 + \cdots + x^{2^{n-1}}$ has $n-1$ additions and $n-1$ squarings in F_{2^n} in $O(n^3)$ time. Thus, we have $Dl^2 \times O(n^3) = O(Dl^2n^3)$ time.
5. $l \times O(n^3) = O(ln^3)$ time.
6. $3 \times Dl^2 + 1 \times l + 1$ ops in F_2. So $O(2^l Dl^2)$ time.

Therefore we have:

$$O(C_{DL}l^2(n^3 + 2^l))$$

time. The input size (number of bits required to represent f) is about $C_{DL}n \log p = C_{DL}n \log 2$ and clearly the complexity does not depend on the degree of $f(x)$ while it primarily depends on the dimension l of the kernel of $T_D(x)$ and the extension degree n of F_{2^n}. □

Note that Algorithm 30 does not resolve the *sign* of the Weil sum S since the returned value is the absolute value $|S|$. We will provide the resolution techniques of the sign of the Weil sum in the next chapter. As is stated in the beginning of this chapter, if central polynomial has a nonzero constant term, we can separately calculate the character value of the constant and multiply by it the result obtained from the Algorithm 30.

In practice, the dimension l of the kernel of the matrix B of $T_D(x)$ is usually small and the experiments in Chapter 6 imply that the parity checking step (6) in Algorithm 30 is very feasible for larger n of our cryptographic interests.

E. Concluding Remarks

The proof method in the simplification procedures of Theorem 25 in Section B is a natural extension of the combined results of Theorem 1.4. [33] and [31]. We showed that at the level of Weil sum values, we can work on the simplified form of central polynomials, in stead of dealing each coefficients appearing in the linearized terms of the central polynomials. The auxiliary linearized polynomial in Section C turns into the index set of partial Weil sum in Algorithm 30 whereby the dimension of its kernel dominates the time complexity of the algorithm.

In fact, it will be shown in Chapter VI that many of the auxiliary linearized polynomials have their kernels of dimensions much smaller than the extension degree n. We do not claim that this algorithm is optimal. It remains an open question to improve the efficiency of this algorithm. The sign that the Algorithm 30 could not resolve will be handled under a certain conditions on the polynomials in the next chapter.

CHAPTER IV

THE NUMBER OF SOLUTIONS OF A BIVARIATE EQUATION FOR DEMBOWSKI-OSTROM POLYNOMIALS

A. Introduction

We will work on the Dembowski-Ostrom polynomial of the form: $\sum_{i=1}^{D} A_i x^{p^{s_i}+1}$ over F_q in Theorem 25. It is easily shown from Theorem 25 that for any Dembowski-Ostrom polynomial over F_q, there exists a simplified Dembowski-Ostrom polynomial of this form whose Weil sum is equivalent to that of the original Dembowski-Ostrom polynomial.

In Section B, we introduce a bivariate polynomial equation associated this simplified Dembowski-Ostrom polynomial. As is in parallel to the standard application of character theory (Chapter 6. [19]), we examine the number of solutions of this bivariate equation. In Section C, we introduce a relation between the number of the solutions and the Weil sum value of the simplified Dembowski-Ostrom polynomials under a certain condition. The actual computation of the number of solutions is performed in Section D by resolving the sign of the Weil sum which the previous Weil sum Algorithm 30 in Chapter III could not resolve for the generic central polynomials.

B. The Bivariate Equation for Dembowski-Ostrom Polynomials

Let $q = p^n$ be the order of F_q when p is either 2 or an odd prime. Consider an arbitrary Dembowski-Ostrom polynomial $f(x) \in F_q[x]$ of form:

$$f(x) = \sum_{i=1}^{D} A_i x^{p^{s_i}+1},$$

where $A_i \in F_q$. This is a simplified central polynomial with $b = 0$ in Theorem 25. So we can call this $f(x)$ a *simplified Dembowski-Ostrom polynomial*. We denote the greatest common divisor of $s_1, \ldots, s_D$ and n by:

$$\delta = \gcd(s_1, \ldots, s_D, n) = (s_1, \ldots, s_D, n).$$

We now introduce a special type of bivariate polynomial equations that is associated with the simplified Dembowski-Ostrom polynomials. More specifically, we consider the bivariate polynomial equation over $F_q \times F_q$:

$$f(x) = y^{p^\delta} - y,$$

whereby the left-hand side is the simplified Dembowski-Ostrom polynomial in $F_q[x]$ while the right-hand side is a *linearized binomial* $y^{p^\delta} - y \in F_q[y]$. We denote the bivariate polynomial by $g(x, y) \in F_q[x, y]$:

$$g(x, y) = f(x) - y^{p^\delta} + y = \sum_{i=1}^{D} A_i x^{p^{s_i}+1} - y^{p^\delta} + y.$$

Henceforth, we are interested in the number of solutions of the bivariate polynomial equation $g(x, y) = 0$ in $F_q \times F_q$ which we denote by:

$$N = N(g(x, y)) = \sharp\{(x, y) \in F_q \times F_q \mid g(x, y) = 0\}.$$

From a simple observation of this bivariate equation, we can easily obtain an estimate of the value of $N(g(x, y))$ under a certain condition. To see this, recall that for integers r, s where s divides r we have:

$$x^r + 1 = (x^s + 1)(x^{(r/s-1)s} - x^{(r/s-2)s} + \ldots - x^s + 1),$$

if r/s is odd; and

$$x^r - 1 = (x^s + 1)(x^{(r/s-1)s} - x^{(r/s-2)s} + \ldots + x^s - 1),$$

if r/s is even. We can obtain the following lemma for the congruential estimate on N.

Lemma 32 *(Batched Solutions. cf. [33]). Let $f(x)$ be a simplified Dembowski-Ostrom polynomial $\sum_{i=1}^{D} A_i x^{p^{s_i}+1}$ over F_q with each s_i/δ being an odd number and $g(x,y) = f(x) - y^{p^\delta} + y$ the bivariate polynomial. Then, the number of solutions $N(g(x,y))$ of the bivariate equation $g(x,y) = 0$ is estimated as:*

$$N \equiv -1 \mod p^\delta + 1.$$

Proof Note that $\delta = (s_1, \ldots, s_D, n)$ by definition and $p^{s_i} + 1$ is divisible by $p^\delta + 1$ since we have

$$p^{s_i} + 1 = (p^\delta + 1)(p^{(s_i/\delta-1)\delta} - p^{(s_i/\delta-2)\delta} + \cdots - p^\delta + 1),$$

for each odd integer s_i/δ. If $(x,y) \in F_q \times F_q$ is a solution of the equation:

$$\sum_{i=1}^{D} A_i x^{p^{s_i}+1} = y^{p^\delta} - y,$$

over $F_q \times F_q$ with nonzero $x \neq 0$, then (wx, y) is also a solution if $w^{p^\delta+1} = 1$. Hence the solutions $(x,y) \in F_q \times F_q$ of the equation for each nonzero x are in some batch $\{(wx, y) \mid w^{p^\delta+1} = 1\}$ of size $p^\delta + 1$. In addition, there are p^δ solutions $(0, y)$ as linearized binomial $y^{p^\delta} - y = 0$ is satisfied by any y in F_{p^δ}. Therefore, denoting the

number of solutions of nonzero x by integer $t \in Z$, we have:

$$\begin{aligned} N &= (p^\delta + 1)t + p^\delta \\ &\equiv p^\delta \mod p^\delta + 1 \\ &\equiv -1 \mod p^\delta + 1, \end{aligned}$$

which is the desired result. □

C. The Number of Solutions and Weil Sum

Assume that $f(x)$ is a simplified Dembowski-Ostrom polynomial $\sum_{i=1}^{D} A_i x^{p^{s_i}+1}$ over F_q and $S = S(A_1, \ldots, A_D)$ the Weil sum. Denote $N = N(g(x, y))$ the number of solutions of bivariate polynomial equation $g(x, y) = f(x) - y^{p^\delta} + y = 0$ in $F_q \times F_q$. Then, we want to discover a certain relation between the Weil sum value S and the number of solutions N. For this purpose, first we need to express N by using the canonical additive character χ_1. Applying the standard application of character sum (cf. Theorem 14) to the bivariate polynomial: $g(x, y) = \sum_{i=1}^{D} A_i x^{p^{s_i}+1} - y^{p^\delta} + y$, we have

$$\begin{aligned} N &= \frac{1}{q} \sum_{w \in F_q} \sum_{x,y \in F_q} \chi_1(w g(x, y)) \\ &= \frac{1}{q} \sum_{w \in F_q} \sum_{x,y \in F_q} \chi_1(w(\sum_{i=1}^{D} A_i x^{p^{s_i}+1} - y^{p^\delta} + y)). \end{aligned}$$

Theorem 13 can also work here to get:

$$\begin{aligned}\chi_1(w(\sum_{i=1}^{D} A_i x^{p^{s_i}+1} - y^{p^\delta} + y)) &= \chi_1(w\sum_{i=1}^{D} A_i x^{p^{s_i}+1}) \sum_{y\in F_q} \chi_1(w(y - y^{p^\delta})))\\ &= \chi_1(w\sum_{i=1}^{D} A_i x^{p^{s_i}+1})\chi_1(w(y - y^{p^\delta})))\\ &= \chi_1(w\sum_{i=1}^{D} A_i x^{p^{s_i}+1})\chi_1(wy)\chi_1(-wy^{p^\delta})\\ &= \chi_1(w\sum_{i=1}^{D} A_i x^{p^{s_i}+1})\chi_1(w^{p^\delta}y^{p^\delta})\chi_1(-wy^{p^\delta})\\ &= \chi_1(w\sum_{i=1}^{D} A_i x^{p^{s_i}+1})\chi_1(y^{p^\delta}(w^{p^\delta} - w)).\end{aligned}$$

Therefore we can simplify N into:

$$N = \sum_{w\in F_q}\sum_{x\in F_q} \chi_1(w\sum_{i=1}^{D} A_i x^{p^{s_i}+1}) \sum_{y\in F_q} \chi_1(y^{p^\delta}(w^{p^\delta} - w)).$$

The inner sum $\sum_{y\in F_q} \chi_1(y^{p^\delta}(w^{p^\delta} - w))$ is zero unless $w^{p^\delta} - w = 0$. Thus it follows that

$$\begin{aligned}N &= q\sum_{w\in F_{p^\delta}}\sum_{x\in F_q} \chi_1(w\sum_{i=1}^{D} A_i x^{p^{s_i}+1})\\ &= q + \sum_{w\in F^*_{p^\delta}}\sum_{x\in F_q} \chi_1(w\sum_{i=1}^{D} A_i x^{p^{s_i}+1}).\end{aligned}$$

This identity implies that there is a relationship between the number of solutions of the bivariate polynomial equation $g(x, y) = 0$ and the Weil sum of $f(x)$, whereby w is multiplied to $f(x)$ inside the argument of character χ_1. In order to further simplify

it, we must impose several conditions on the types of simplified Dembowski-Ostrom polynomial $f(x) = \sum_{i=1}^{D} A_i x^{p^{s_i}+1}$. Note that δ is defined as the greatest common divisor of n and $s_1, \ldots, s_D$ appearing in the exponents of the monomials in $f(x)$. Although the integers s_i, $1 \leq i \leq D$ are regarded as the exponents in the simplified Dembowski-Ostrom polynomial under these conditions, we have a general lemma for any integer s_i from elementary number theory.

Lemma 33 *(GCD and Exponents. cf. Theorems 1.5. and 1.6 [33]). Let n and s_i, $1 \leq i \leq D$ be the nonnegative integers with a greatest common divisor $\delta = \gcd(s_1, \ldots, s_D, n)$. Note that the condition that s_i/δ is odd for each $1 \leq i \leq D$ forces s_i to be positive. Assume that n/δ is even and s_i/δ is odd for each i. Then, we have:*

$$(\frac{2^{s_i}+1}{2^{\delta}+1}, 2^{\delta}-1) = 1.$$

Proof Note that $U_i = (2^{s_i}+1)/(2^{\delta}+1)$ for each i and $2^{\delta}-1$ are odd, thus $l = \gcd(U_i, 2^{\delta}-1)$ must be odd. Suppose that $l \geq 3$. Since s_i/δ is odd for each i, we have:

$$2^{s_i}+1 = (2^{\delta}+1)(2^{(s_i/\delta-1)\delta} - 2^{(s_i/\delta-2)\delta} + \ldots - 2^{\delta} + 1),$$

which implies:

$$U_i = 2^{(s_i/\delta-1)\delta} - 2^{(s_i/\delta-2)\delta} + \ldots - 2^{\delta} + 1.$$

By the definition of l, the odd number l divides both U_i and $2^{\delta}-1$, hence l divides the sum:

$$\begin{aligned} U_i + (2^{\delta}-1) &= 2^{(s_i/\delta-1)\delta} - 2^{(s_i/\delta-2)\delta} + \ldots - 2^{3\delta} + 2^{2\delta} \\ &= 2^{2\delta}(2^{(s_i/\delta-3)\delta} - 2^{(s_i/\delta-4)\delta} + \ldots - 2^{\delta} + 1). \end{aligned}$$

Because $l \geq 3$ is odd, it again divides the second factor $(2^{(s_i/\delta-3)\delta} - 2^{(s_i/\delta-4)\delta} + \ldots -$

$2^\delta + 1)$ in the sum $U_i + (2^\delta - 1)$. By repeating this process, l must eventually divide

$$(2^{2\delta} - 2^\delta + 1) + (2^\delta - 1) = 2^\delta,$$

which is a contradiction. (This proof induces the contradiction slightly different from the proof of Lemma 3.6. in [33]). □

Now, we can complete the task left in Lemma 3.6 in [33] for the case $p = 2$. That is, we can combine the previously known result when p is an odd prime in [33] to the case $p = 2$. The following theorem unifies the case $p = 2$ and the case that p is an odd prime.

Theorem 34 *(Bivariate Equation. Emulation Condition. Lemma 3.6. [33]). Let F_q be a finite field of order $q = p^n$, where p is the characteristic, and $f(x)$ a Dembowski-Ostrom polynomial $\sum_{i=1}^{D} A_i x^{p^{s_i}+1}$ over F_q. Denote by $N(g(x, y))$ the number of solutions of bivariate equation $g(x, y) = f(x) - y^{p^\delta} + y$ over $F_q \times F_q$. Suppose that with $\delta = (s_1, \ldots, s_D)$:*

1. *n/δ is even,*
2. *$\delta = (s_i, n)$ for each i,*
3. *s_i/δ is odd for each i, and*
4. *2δ divides $s_i - s_j$ for all $i \neq j$.*

Then, we have:

$$N(g(x, y)) = q + (p^\delta - 1)S,$$

where $S = S(A_1, \ldots, A_D) = \sum_{x \in F_q} \chi_1(\sum_{i=1}^{D} A_i x^{p^{s_i}+1})$: the Weil sum of $f(x)$.

Proof Since "odd" case is proved in [33], we only show the proof for $p = 2$. Let us

start with the identity obtained in the previous discussion:

$$N = 2^n + \sum_{w \in F^*_{p^\delta}} \sum_{x \in F_q} \chi_1(w \sum_{i=1}^{D} A_i x^{2^{s_i}+1}).$$

For the integers s_i's appearing in the exponents of the polynomial, by Lemma 33 we have:

$$(\frac{2^{s_i}+1}{2^\delta+1}, 2^\delta - 1) = 1$$

for each i. Multiplying both elements $(2^{s_i}+1)/(2^\delta+1)$ and $2^\delta - 1$ by $2^\delta + 1$ yields

$$(2^{s_i}+1, 2^{2\delta} - 1) = 2^\delta + 1.$$

Since n/δ is even, we have:

$$2^n - 1 = (2^\delta + 1)(2^{(n/\delta-1)\delta} - 2^{(n/\delta-2)\delta} + \ldots + 2^\delta - 1)$$

and the fact that $(2^\delta - 1, 2^\delta + 1) = 1$ by Lemma 3 (i.e., $2^\delta - 1$ and $2^\delta + 1$ are relatively prime). So $2^\delta + 1$ must divide $(2^n - 1)/(2^\delta - 1)$. Let g be an arbitrary primitive element of F_q (Definition 9). We first want to show that for each i and for any $w \in F^*_{p^\delta}$, the equation:

$$w z_w^{2^{s_i}+1} = 1$$

is solvable for z_w in $F_{2^{2\delta}}$. Let us express any $w \in F^*_{2^\delta}$ for some integer s as:

$$w = g^{s(\frac{q-1}{2^\delta-1})}$$

and set the unknown z_w to be solved in $F_{2^{2\delta}}$(obviously z_w is dependent on w) as:

$$z_w = g^r,$$

with some integer r. Consequently, we need to consider the solvability of the equation:

$$g^{s(\frac{q-1}{2^\delta-1})} g^{r(2^{s_i}+1)} = 1,$$

for the unknown integer r. Equivalently, we must examine the corresponding linear congruence equation:

$$r(2^{s_i}+1) \equiv -s\frac{q-1}{2^\delta-1} \mod 2^{2\delta}-1.$$

From Lemma 4, this is solvable for r if and only if

$$(2^{s_i}+1, 2^{2\delta}-1) \text{ divides } -s\frac{q-1}{2^\delta-1}.$$

Since $(2^{s_i}+1, 2^{2\delta}-1) = (2^\delta+1)$, the linear congruence equation is solvable for r if and only if:

$$(2^\delta+1) \text{ divides } -s\frac{q-1}{2^\delta-1},$$

which is true regardless of the value of s as $(2^\delta+1)$ readily divides $(q-1)/(2^\delta-1)$. (The reader may finally notice why z_w is sought in $F_{2^{2\delta}}$ of size $2^{2\delta}$.) Therefore, we can show that for each i and for any $w \in F^*_{p^\delta}$, an equation $wz_w^{2^{s_i}+1} = 1$ is solvable for z_w in $F_{2^{2\delta}}$.

Next, we need to show that the $z_w \in F_{2^{2\delta}}$ in $wz_w^{2^{s_i}+1} = 1$ for some i $(1 \le i \le D)$ also satisfies $wz_w^{2^{s_j}+1} = 1$ for any $j \neq i$. In other words, for an arbitrary i, we want to know whether:

$$wz_w^{2^{s_i}+1} = 1 \text{ for some } z_w \in F_{2^{2\delta}} \Rightarrow wz_w^{2^{s_j}+1} = 1 \text{ for any } j \neq i.$$

Assume that this is true for arbitrarily fixed i. We have the solvable equation $wz_w^{2^{s_i}+1} = 1$ with some $z_w \in F_{2^{2\delta}}$. By assumption 2δ divides $s_j - s_i$, so set some $k \in Z$ such that $s_j - s_i = 2k\delta$. Then, for any $j \neq i$ and the solution $z_w \in F_{2^{2\delta}}$ of

$wz_w^{2^{s_i}+1} = 1$, we have:

$$wz_w^{2^{s_j}+1} = z_w^{-(2^{s_i}+1)} z_w^{2^{s_j}+1} = z_w^{2^{s_j}-2^{s_i}} = (z_w^{2^{s_j-s_i}})^{2^{s_i}}.$$

Without loss of generality we assume k is positive. When $k \geq 2$, we recursively have:

$$z_w^{2^{s_j-s_i}} = z_w^{2^{2\delta k}} = z_w^{2^{2\delta(k-1)}} \times \cdots \times z_w^{2^{2\delta(k-1)}} (2\delta \text{ times}),$$

while the case $k = 1$ yields:

$$z_w^{2^{s_j-s_i}} = z_w^{2^{2\delta}} = 1.$$

Therefore we have shown that:

$$wz_w^{2^{s_j}+1} = 1.$$

Finally, this common solution $z_w \in F_{2^{2\delta}}^*$ for all the equations $wz_w^{2^{s_i}+1} = 1$ for $1 \leq i \leq D$ can further simplify N as follows.

$$\begin{aligned}
N &= 2^n + \sum_{w \in F_{2^\delta}^*} \sum_{x \in F_{2^n}} \chi_1(w(\sum_{i=1}^{D} A_i x^{2^{s_i}+1})) \\
&= 2^n + \sum_{w \in F_{2^\delta}^*} \sum_{z_w x \in F_{2^n}} \chi_1(\sum_{i=1}^{D} A_i w (z_w x)^{2^{s_i}+1}) \\
&= 2^n + \sum_{w \in F_{2^\delta}^*} \sum_{z_w x \in F_{2^n}} \chi_1(\sum_{i=1}^{D} A_i (w z_w^{2^{s_i}+1}) x^{2^{s_i}+1}) \\
&= 2^n + \sum_{w \in F_{2^\delta}^*} \sum_{z_w x \in F_{2^n}} \chi_1(\sum_{i=1}^{D} A_i x^{2^{s_i}+1}) \\
&= 2^n + (2^\delta - 1) S(A_1, \ldots, A_D).
\end{aligned}$$

Therefore, we obtained the desired result. □

The fact that adversary may know the value N of bivariate equation $\sum_{i=1}^{D} A_i x^{2^{s_i}+1} = y^{2^\delta} - y$ has some implications when $f(x)$ is used in multivariate quadratic trapdoor

function. To see this assume that $(x, y) \in F_q \times F_q$ is a solution to the bivariate equation. Then, the mapping $x \mapsto f(x)$ which is supposed to be one-way thus hard to invert, can be "emulated" by the linearized mapping $y \mapsto y^{2^\delta} - y$ which is easy to invert. This observation will be integrated into the attack on trapdoor based on these forms of D-O polynomials in next section.

In this paper, the conditions in Theorem 34 on the extension degree n of F_q and the integers $s_1, \ldots, s_D$ appearing in the portions of the exponents of the simplified D-O polynomial are called the *emulation conditions* for this cryptanalytic reason. That is, $f(x) = \sum_{i=1}^{D} A_i x^{p^{s_i}+1}$ over F_{p^n} in Theorem 26 satisfies the emulation conditions if with $\delta = (s_1, \ldots, s_D)$:

$$\begin{cases} n/\delta \text{ is even,} \\ \delta = (s_i, n) \text{ for each } i, \\ s_i/\delta \text{ is odd for each } i, \text{and} \\ 2\delta \text{ divides } s_i - s_j \text{ for all} j \neq i. \end{cases}$$

D. Computation of the Number of Solutions

It is important to note that we do not have an analogue of Theorem 1.4 [33] when $p = 2$. On the other hand, when p is an odd prime, there is neither the convenient lemma such as Lemma 28 nor the straightforward proof such as in the subsequent Corollary 29 that is applicable even to the simplified Dembowski-Ostrom polynomial as a central polynomial.

Although we do not need the results of Theorem 1.4 in [33] as they are inapplicable to the Weil sum $S(A_1, \ldots, A_D)$ of $p = 2$, we still want to detail the interesting proof techniques used in the theorems of [33] by highlighting the fact that the results of Theorem 1.4 are apparently inapplicable to our new case $p = 2$.

Corollary 35 *(Inapplicability). Let p be any prime (2 or odd prime) and F_q a finite field of order $q = p^n$. Let $f(x)$ be a simplified Dembowski-Ostrom polynomial $f(x) = \sum_{i=1}^{D} A_i x^{p^{s_i}+1})$ and $T_D(x)$ the auxiliary linearized polynomial $T_D(x) = A_1^{p^{s1}} x^{p^{2s_1}} + A_1 x + \sum_{i=2}^{D} [A_i^{p^{s1}} x^{p^{s_1+s_i}} + (A_i x)^{p^{s_1+y_i}}]$, as is in Theorem 26. Then, for any root $w \in F_q$ of the equation $T_D(w) = 0$ we have:*

$$\chi_1(\sum_{i=1}^{D} A_i w^{p^{s_i}+1}) = \overline{\chi_1}(\sum_{i=1}^{D} A_i w^{p^{s_i}+1}).$$

Proof Since the proof scheme in Mills' theorem is also applicable to our new case $p = 2$, we only highlight the portions of the entire proof in [33] which are not explicitly described in the paper. Assume that $D \geq 2$. We handle the terms $i = 1, 2$ separately from the rest terms $i \geq 3$ in formula (7) of Mills' proof. First we have:

$$\prod_{i=1}^{D} \chi_1(A_i w^{p^{s_i}+1}) = \prod_{i\neq 2}^{D} \chi_1(A_i w^{p^{s_1}+1}) \chi_1([A_2^{p^{s_1}} w^{p^{s_1+s_2}} - T_D(w)] w^{p^{s_1}}).$$

For the term $i = 1$ of T_D on the right-hand side of the identity , we can simplify as:

$$\begin{aligned} \chi_1(A_1 w^{p^{s_1}+1} - [A_1^{p^{s_1}} w^{p^{2s_1}} + A_1 w] w^{p^{s_1}}) &= \chi_1(-A_1^{p^{s_1}} (w^{p^{s_1}})^{p^{s_1}} w^{p^{s_1}}) \\ &= \overline{\chi_1}(A_1^{p^{s_1}} w^{p^{s_1}+1}). \end{aligned}$$

For the term $i = 2$, we can simplify as:

$$\chi_1(-(A_2 w)^{p^{s_1+y_2}} w^{p^{s_1}}) = \overline{\chi_1}(A_2^{p^{y_2}} w^{p^{y_2}+1}).$$

And for terms $i \geq 3$, we can simplify as:

$$\chi_1(A_i w^{p^{s_i}+1} - (A_i^{p^{s_1}} w^{p^{s_1+s_i}} + (A_i w)^{p^{s_1+y_i}}) w^{p^{s_1}}) = \overline{\chi_1}(A_i^{p^{y_i}} w^{p^{y_i}+1}).$$

Since we have:

$$\overline{\chi_1}((A_i^{p^{y_i}} w^{p^{y_i}+1})^{p^{s_i}}) = \overline{\chi_1}(A_i w^{p^{s_i}+1})$$

for each $i \geq 2$, we have:

$$\chi_1(\sum_{i=1}^{D} A_i w^{p^{s_i}+1}) = \overline{\chi_1}(\sum_{i=1}^{D} A_i w^{p^{s_i}+1}).$$

□

When p is odd, both sides of the obtained equality in Corollary 35 (also in Theorem 1.4 [33]) are easily shown to be 1. To see this, consider:

$$\exp(\frac{2\pi i}{p} Tr(\sum_{i=1}^{D} A_i w^{p^{s_i}+1})) = \exp(-\{\frac{2\pi i}{p} Tr(\sum_{i=1}^{D} A_i w^{p^{s_i}+1})\}),$$

where w is a root of $T_D(w) = 0$. Since both sides (complex conjugates) are equal, they must be *real* i.e. must be 1 or -1. In addition, $Tr(\sum_{i=1}^{D} A_i w^{p^{s_i}+1})$ must be divisible by odd denominator p, which makes both sides to be 1. If Theorem 26 is applied to this simplified Dembowski-Ostrom polynomial, we have:

$$|S|^2 = q \sum_{T_D(w)=0, w \in F_q} \chi_1(\sum_{i=1}^{D} A_i w^{p^{s_i}+1}) = q \sum_{T_D(w)=0, w \in F_q} 1 = qN(T_D(x) = 0),$$

and thus we obtain the actual result of Theorem 1.4 of [33].

However, when $p = 2$, we do not have to examine if character $\chi_1(f(x))$ is real or not, but the value of $\chi_1(\sum_{i=1}^{D} A_i w^{2^{s_i}+1})$ can actually *flip* between 1 and -1. In other words, while $\chi_1(f(x))$ is always guaranteed to be real by Lemma 28, we cannot get any analogous convenient result such as $|S|^2 = 2^n N(T_D = 0)$ of Theorem 1.4 in [33] if $p = 2$.

Let us end this chapter by providing a customized version of the Weil sum algorithm Algorithm 30 by resolving the sign of the absolute value computed in the algorithm. Since Weil sum algorithm, Algorithm 30, works on the case $p = 2$, we

apply it to the simplified Dembowski-Ostrom polynomial in Theorem 34 to obtain the Weil sum with $p = 2$. Unlike the case where p is an odd prime [30, 31, 33], we have a very simple sign resolution scheme for our case $p = 2$.

Theorem 36 *(Sign of Weil Sum). Let F_q be a finite fields of order $q = 2^n$, where 2 is the characteristic, and $f(x)$ a simplified Dembowski-Ostrom polynomial $\sum_{i=1}^{D} A_i x^{2^{s_i}+1}$ over F_{2^n} whereby the emulation conditions in Theorem 34 are satisfied. Also let $S = S(A_1, \ldots, A_D) = \sum_{x \in F_{2^n}} \chi_1(\sum_{i=1}^{D} A_i x^{2^{s_i}+1})$ be the Weil sum of $f(x)$ while $|S|$ denote its absolute value calculated by the Weil sum algorithm Algorithm 30. Then, we have:*

$$S = \begin{cases} +|S| & \text{if } 2(1-|S|) \equiv 0 \mod 2^\delta + 1, \\ -|S| & \text{otherwise.} \end{cases}$$

Proof For the simplified Dembowski-Ostrom polynomial $f(x)$, we have an identity from Theorem 34:

$$N = q + (2^\delta - 1)S,$$

where $N = N(f(x), y^{2^\delta} - y)$ is the number of the solutions of the bivariate equation $f(x) = y^{2^\delta} - y$ in F_q. From Lemma 32, N can be congruently estimated as:

$$N \equiv -1 \mod 2^\delta + 1.$$

By combining both identities, we have:

$$0 \equiv N + 1 \equiv 2(1 - S) \mod 2^\delta + 1.$$

Now, we want to determine the sign of $S = \pm|S|$. Assume that S has a sign $+$, i.e., $S = |S|$. Then we have the congruence identity:

$$0 \equiv N + 1 \equiv 2(1 - |S|) \mod 2^\delta + 1.$$

In this case, the absurdity of assuming $-$ sign for $S = -|S|$ is easily shown because of the other congruence identity:

$$0 \equiv N + 1 \equiv 2(1 - (-|S|)) \mod 2^{\delta} + 1$$

yields a contradiction in the sum with the original identity as:

$$4 \equiv 0 \mod 2^{\delta} + 1.$$

Therefore, $S = |S|$ if and only if $0 \equiv 2(1 - |S|) \mod 2^{\delta} + 1$, and we obtained the desired result. □

Now we can add the result of this sign resolution scheme of Theorem 36 into the Weil sum algorithm Algorithm 30. In this case the input of the algorithm must be a simplified Dembowski-Ostrom polynomial with the emulation conditions in Theorem 34.

Algorithm 37 *(Simplified Weil Sum Algorithm. $p = 2$).*

INPUT $f(x) = \sum_{i=1}^{D} A_i x^{2^{s_i}+1}$*: Dembowski-Ostrom polynomial over* F_{2^n} *with the emulation conditions in Theorem 34.*

OUTPUT *S: Weil sum of* $f(x)$*.*

1. *Compute the associated linearized* $T_D(x) \in F_{2^n}$ *as in Theorem 26 (Suppose* l *is the dimension of* $\ker(T_D)$*).*
2. *Compute the basis* $\{\eta_1, \ldots, \eta_l\}$ *of* $\ker(T_D)$*.*
3. *Let* U *be* $0 \in Z$*.*
4. *Compute* $\gamma_{i,j_1,j_2} = Tr(A_i \eta_{j_1} \eta_{j_2}^{2^{s_i}})$ *for* $1 \leq i \leq D, 1 \leq j_1, j_2 \leq l$

5. *For each* $(x_1, \ldots, x_l) \in F_2^l$, *evaluate*

$$C_{(x_1,\ldots,x_l)} = \sum_{i=1}^{D} \sum_{j_1=1}^{l} \sum_{j_2=1}^{l} x_{j_1} x_{j_2} \gamma_{i,j_1,j_2} \in F_2$$

and set $U = U + C_{(x_1,\ldots,x_l)}$. *(Note: integer addition.)*

6. *If* $|S| = 2^{n/2}\sqrt{2^l - 2U}$ *satisfies* $2(1 - |S|\ |) \equiv 0 \mod 2^\delta + 1$ *then return* $|S|$; *otherwise return* $-|S|$.

The time complexity of this modified algorithm is: $O(Dl^2(n^3 + 2^l))$ time in terms of the binary operations on RAM. This complexity is similar to that of Theorem 31.

E. Concluding Remarks

Lemma 32 of the batched solutions require only s_i/δ to be odd in the emulation conditions for the Dembowski-Ostrom polynomials. All the known methods of the explicit Weil sum evaluation of Dembwoski-Ostrom polynomial [30, 31, 32, 33] carefully separate the four combinations of evenness and oddness for characteristic p and n/δ. This is mainly because of the subtleties in Lemma 3 regarding the greatest common divisors.

As is open in [33] for the case where p is an odd prime, it remains an open question to extend the emulation conditions proposed in Section C to the case n/δ is odd for $p = 2$. The sign resolution in Theorem 36 contrasts to that of others when p is odd. This sign resolution enabled us to obtain the concrete value of the Weil sum of the Dembowski-Ostrom polynomials under the emulation conditions in Algorithm 37.

CHAPTER V

SECURITY OF A CLASS OF DEMBOWSKI-OSTROM POLYNOMIALS

A. Introduction

In this chapter, we investigate what kind of mechanism in the cryptosystem design could potentially affect the security of the multivariate quadratic cryptosystem.

In Section B, we discuss an attack on the short signature schemes based on the birthday problem in Definition 7. This attack imposes a lower bound of the size of a cryptosystem design in order to achieve the commonly prescribed security level of 2^{80}. In Section C, we introduce a new customization of the attack in the previous section on the Demboswki-Ostrom polynomials with emulation conditions in the generic MQ-trapdoor. It is shown that the new attack could be asymptotically better than the attack based on the birthday problems for infinitely many possible extension degree n so that we can characterize the nontrivial class of weak Dembowski-Ostrom polynomials in the MQ-trapdoor under this new attack.

B. Generic Threats against Digital Signature Scheme

Let F_p and F_q be the finite fields with $q = p^n$. As usual the extension field F_q is regarded as a n-dimensional vector space F_p^n over F_p with some fixed basis. We consider a MQ problem $MQ(p, n, n)$ (Definition 18) of the system of n multivariate quadratic polynomial equations $z = F(x) = (P_1(x_1, \ldots, x_n), \ldots, P_n(x_1, \ldots, x_n))$ s.t.

$$z_k = P(x_1, \ldots, x_n) = \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_{i,j}^{(k)} x_i x_j + \sum_{i=1}^{n} \beta_i^{(k)} x_i + \gamma^{(k)},$$

where $\alpha_{i,j}^{(k)}, \beta_i^{(k)}, \gamma^{(k)} \in F_p$ for $1 \leq k \leq n$. We also assume that some generic trapdoor structure is embedded into this system so that the system $F = (P_1(x_1, \ldots, x_n), \ldots,$

$P_n(x_1, \ldots, x_n))$ is a public mapping (public key) $F_p^n \to F_p^n$ and forms a $MQ(p, n, n)$-trapdoor. The actual inversion of the mapping F is feasible only with the knowledge of the secret functional composition of a central polynomial over F_q and two affine bijections $F_p^n \to F_p^n$ (cf. HFE-trapdoor, Definition 23).

By Lemma 19, there exists a univariate representation over F_q of the $MQ(p, n, n)$-trapdoor whose form is identical to that of the central polynomial (Definition 21)

$$f(x) = \sum_{i=1}^{D} a_i x^{p^{\alpha_i}+p^{\beta_i}} + \sum_{j=1}^{L} b_j x^{p^{\gamma_j}} + c,$$

where $a_i, b_i, c \in F_q$. If the system is homogeneous, we have a unique Dembowski-Ostrom polynomial (Definition 22):

$$f(x) = \sum_{i=1}^{D} a_i x^{p^{\alpha_i}+p^{\beta_i}},$$

where $a_i \in F_q$ to express this $MQ(p, n, n)$-trapdoor.

Consider $MQ(p, n, n)$-trapdoor that is used for a digital signature scheme. While the verification of a signature $x = (x_1, \ldots, x_n) \in F_p^n$ is performed with the public key, $F(x) = (P_1(x_1, \ldots, x_n), \ldots, P_n(x_1, \ldots, x_n))$, the signing of a (hashed) message value $(z_1, \ldots, z_n) \in F_p^n$ (often with some redundancy) requires knowledge of the trapdoor information. In a digital signature scheme, the goal of an adversary is to *forge* signatures; that is, produce signatures which will be accepted as those of some other entity. One of the weakest forms of forgeries in digital signature scheme called *existential forgery* in the following.

Definition 38 *(Existential Forgery, Section 11.24. [2]). An adversary is able to forge a signature of at least one message over which the adversary has little or no control.*

In an existential forgery of digital signature scheme, an attack is considered to be successful if he can either deterministically or probabilistically counterfeit at least

one valid signature of some message.

With regard to the types of information and access available to the adversary, we can categorize the possible modes of attacks.

Definition 39 *(Modes of Attacks. Section 11.2.4. [2]) There are two basic attacks against public-key digital signature scheme.*

Key-only attack: *An adversary knows only the signer's public key.*

Message attack: *An adversary is able to examine signatures corresponding either to known or chosen messages. Message attacks can be further subdivided into three classes.*

1. *Known-message attack: An adversary has signatures for a set of messages which are known to the adversary but not chosen by him.*
2. *Chosen-message attack: An adversary obtains valid signatures form a chosen list of messages before attempting to break the signature scheme. This attack is non-adaptive in the sense that messages are chosen before any signatures are seen.*
3. *Adaptive chosen-message attack: An adversary is allowed to use the signer as an oracle. The adversary may request signatures of messages which depend on the signer's public key; and he may request signatures of messages which depend on previously obtain signatures or messages.*

In general, an adaptive chosen message attack is one of the strongest modes of attacks on digital signature scheme; and any practical digital signature scheme must possess (either provably or computationally suggestive) substantial security against this type of attack. A *security parameter* is often defined as the (minimum) computational complexity that the cryptosystem must impose on any possible attacker in

order to achieve the prescribed security objective. It is currently considered that any practically useful cryptosystem should have security parameter at least 2^{80} (cf. [3]).

Among the currently known multivariate quadratic schemes which offer very short signature size, the possibility of a *birthday attack* is often considered when selecting the proper size of cryptosystem (in our case, the degree n of extension field F_{p^n}).

Definition 40 *(Birthday Attack. cf. [20]). Let*

$$F(x) = (P_1(x_1, \ldots, x_n), \ldots, P_n(x_1, \ldots, x_n))$$

be a system of n multivariate polynomials over F_p which represents the corresponding $MQ(p, n, n)$-trapdoor $F : F_p^n \to F_p^n$.

A birthday attack on digital signature scheme based on $MQ(p, n, n)$-trapdoor is an existential forgery such that an adversary:

1. *Generates $p^{n/2}$ random signature values σ_i and sorted list $\{F(\sigma_1), F(\sigma_2), \ldots, F(\sigma_{p^{n/2}})\}$,*
2. *Generates $p^{n/2}$ random message values m_j and sorted list of their hash values $\{z_1, z_2, \ldots, z_{p^{n/2}}\}$, and then*
3. *Searches for a coincidence $F(\sigma_i) = z_j$ for some pair $(i, j) \in [1, p^{n/2}]^2$.*

The time complexity of such an attack is $O(p^{n/2})$.

In a birthday attack, the adversary generates $p^{n/2}$ messages for each list. This is because from the birthday problem Definition 7 the expected number of the elements to select from F_q of order $q = p^n$ while having no coincidence is approximately:

$$\sqrt{\frac{\pi}{2} p^n} \approx p^{n/2}.$$

Therefore, after $p^{n/2}$ messages are selected, it is expected that the adversary can find out at least one coincidence $F(\sigma_i) = z_j$ for some (i, j) in the two lists.

In short signature schemes such as the ones based on HFE-trapdoor or TTS-trapdoor, n is usually chosen $n > 160$ with $p = 2$ because of the possibility for this birthday attack with complexity $2^{n/2} = 2^{80}$. The birthday attack is a generic attack on any digital signature scheme based on $MQ(p, n, n)$-trapdoor, and it does not provide any insight into the weak key structures which may exist inside the key selection mechanism of the cryptosystem.

In fact, as is the case for many other known attacks based on equation-solving such as Gröbner basis methods [6], the underlying problem that the birthday attacker attempts is to solve is also the system of the multivariate quadratic equations. In other words, from the two lists $\{F(\sigma_1), F(\sigma_2), \ldots, F(\sigma_{p^{n/2}})\}$ and $\{z_1, z_2, \ldots, z_{p^{n/2}}\}$, the birthday attack seeks for at least one *solution* of some equation:

$$F(x) = z,$$

with the two variables $x, z \in F_p^n \times F_p^n$. This crucial fact that a birthday attack is equivalently seeking for at least one *root* of the bivariate polynomial $g(x, z) = f(x) - z$ over $F_q \times F_q$ is the primary motivation for considering our weak key identification scheme in the following sections.

It is also interesting to note that, in contrast to many other algebraic attacks, the time complexity of birthday attack does not depend on the degree of $f(x)$ in the univariate form or other intermediate polynomials during the computation, which greatly contrasts to those of many other algebraic attacks whose time complexity often depends on the degree of the polynomials.

C. A Class of Weak Dembowski-Ostrom Polynomials

From the previous observation it is shown that the birthday attacker actually seeks for a solution of the corresponding bivariate equation over $F_q \times F_q$:

$$g(x, z) = f(x) - z,$$

where $f(x) = \sum_{i=1}^{D} a_i x^{p^{\alpha_i}+p^{\beta_i}} + \sum_{j=1}^{L} b_j x^{p^{\gamma_j}} + c$ is a central polynomial. Our goal is to develop a systematic detection method for some class of the Dembowski-Ostrom polynomials of simplified form (Theorem 34):

$$f(x) = \sum_{i=1}^{D} A_i x^{p^{s_i}+1}$$

whose security is less than the birthday attack security parameter $\sqrt{\frac{\pi}{2}p^n} \approx p^{n/2}$ in Definition 40. Let us now start to characterize this class of the weak polynomial structures.

1. Number of Dembowski-Ostrom Polynomials

Let $f(x)$ be a secret Dembowski-Ostrom polynomial:

$$f(x) = \sum_{i=1}^{D} A_i x^{p^{s_i}+1} = A_1 x^{p^{s_1}+1} + \cdots + A_D x^{p^{s_D}+1},$$

where $D \geq 1$, $A_i \in F_q$ for $1 \leq i \leq D$ and $0 \leq s_1 < s_2 < \ldots < s_D \leq q-1$. It is easy to show that there are $\frac{q^D q^{(D)}}{D!}$ simplified Dembowski-Ostrom polynomial in $MQ(p, n, n)$-trapdoor. Henceforth, we consider $MQ(2, n, n)$ of the case $q = 2^n$.

Among these simplified Dembowski-Ostrom polynomials, the crucial conditions for the s_i's in Theorem 34 are applied to determine the scope of the subset $\mathcal{K}$. That is, we consider a class of simplified Dembowski-Ostrom polynomials: with $D \geq 1$ and

$\delta = (s_1, \ldots, s_D, n)$ such that:

$$\mathcal{K} = \{\sum_{i=1}^{D} A_i x^{2^{s_i}+1}, n/\delta \text{ even}, \delta = (s_i, n), s_i/\delta \text{ odd}, 2\delta \text{ divides } s_i - s_j\}.$$

This subset $\mathcal{K}$ is the set of Dembowski-Ostrom polynomials among all the possible central polynomials in $MQ(2, n, n)$-trapdoors from which we attempt to identify the polynomials (asymptotically) weaker than birthday attack security parameter.

2. Linearized Binomial Attack on Dembowski-Ostrom Polynomials

In order to define the class of Dembowski-Ostrom polynomials in $\mathcal{K}$ weaker than birthday attack parameter, we must determine an attack which is stronger than the birthday attack against the polynomials. Suppose that $f(x)$ is a Dembowski-Ostrom polynomial in $\mathcal{K}$, which is built into $MQ(2, n, n)$-trapdoor. Now, consider a scenario in which the number of solutions N of the associated bivariate equation:

$$f(x) = y^{2^\delta} - y,$$

over $F_{2^n} \times F_{2^n}$ is substantially large for the mapping $f : F_{2^n} \to F_{2^n}$. As is shown in Lemma 32, if $x \neq 0$ is a nonzero solution $(x, y) \in F_{2^n} \times F_{2^n}$ with some y, then for any $w \in F_{2^n}$ with $w^{2^\delta+1} = 1$, (wx, y) is also a solution. In other words, each nonzero $x \neq 0$ of solution (x, y) resides in the batch of each size $2^\delta + 1$ in F_{2^n}.

In fact, it is followed that the partition by these batches induces an equivalence relation in $F_{2^n}^*$. I.e. for any $x, x' \in F_{2^n}^*$ (i.e., nonzero solutions),

$$x \approx x' \iff \exists w \text{ s.t. } x' = wx, \text{ and } w^{2^\delta+1} = 1.$$

Here, note that since n/δ is even, $2^\delta + 1$ divides $2^n - 1$. Therefore, we can obtain the

set of equivalence classes: $B_1, B_2, \ldots, B_T$ of each size $2^\delta + 1$ in $F_{2^n}^*$ such that:

$$B_i \cap B_j = \emptyset, \text{ and } \bigcup_{1 \leq i \leq T} B_i = F_{2^n}^*,$$

with $1 \leq i < j \leq T$ and $T = \frac{2^n - 1}{2^\delta + 1}$.

Assume that $t \in Z$ is the number of the classes B_i in $F_{2^n}^*$ such that any element of the classes are actually the solutions of the bivariate equation; that is, there exist t equivalent classes:

$$B_{i_1}, B_{i_2}, \ldots, B_{i_t}$$

whose elements are the solutions of $f(x) = y^{2^\delta} - y$. If we randomly pick up an arbitrary element x from $F_{2^n}^*$, x may belong to some class B_i of the solutions at probability $\frac{t}{T}$. Therefore, after approximately $\frac{T}{t}$ random choices, it is expected that we can pick up at least one solution x in some solution class B_i. Note that for this $x \in B_i$, we have:

$$f(x) \in Im(L),$$

where $Im(L)$ is the image of mapping $L : y \mapsto y^{2^\delta} - y$ over F_{2^n}. In other words, after T/t random selection of $x \in F_{2^n}^*$, we will obtain at least one element $f(x)$ that is located inside $Im(L)$. Since $L(y) = y^{2^\delta} - y$ is a linearized polynomial with kernel of size 2^δ, the image $Im(L) \subseteq F_{2^n}$ is of size $2^{n-\delta}$. Therefore, it follows that after approximately T/t random generations of values $f(x)$, we will have some element $f(x)$ in F_{2^n} which is also inside $Im(L)$.

Now, we ready for applying the birthday attack in Definition 40 on this range $Im(L)$ of size $2^{n-\delta}$ in F_{2^n}. Assume that the attack seeks the nonzero solutions $x \neq 0$ of $(x, y) \in F_q \times F_q$ by working on the secret simplified Dembowski-Ostrom polynomial $f(x) \in \mathcal{K}$ which represents the $MQ(2, n, n)$-trapdoor $F(x) = (P_1(x_1, \ldots, x_n), \ldots, P_n(x_1, \ldots, x_n))$.

Definition 41 *(Linearized Binomial Attack. Univariate Form). Let F_{2^n} be a finite field with even extension degree n with some fixed basis as vector space F_2^n. Suppose $f(x)$ is an arbitrary simplified Dembowski-Ostrom polynomial in $\mathcal{K}$ under the emulation conditions in Theorem 34 which expresses the corresponding $MQ(2,n,n)$-trapdoor $F(x) = (P_1(x_1,\ldots,x_n),\ldots,P_n(x_1,\ldots,x_n))$ over F_2^n. Then, a linearized binomial attacker performs the following:*

1. *Randomly guess the value of the unknown $\delta = (s_1,\ldots,s_D,n)$ from $\{1,\ldots,n\}$. This δ allows the adversary to fix a linearized binomial $L(y) = y^{2^\delta} - y$ in $F_q[y]$. We denote by $Im(L)$ the image of the mapping L over F_q.*
2. *Generate $\frac{T}{t}2^{\frac{n-\delta}{2}}$ random elements $x \in F_q$ and obtain the list:*
$$\{f(x_1), f(x_2), \ldots, f(x_{\frac{T}{t}2^{\frac{n-\delta}{2}}})\}.$$
3. *Generate $2^{\frac{n-\delta}{2}}$ random elements $z \in Im(L)$ to obtain the list:*
$$\{z_1, \ldots, z_{2^{\frac{n-\delta}{2}}}\}.$$
4. *Search for a coincidence $f(x_j) = z_i$ for some i, j in the two lists.*

To analyze how the *linearized binomial attack* works on the simplified Dembowski-Ostrom polynomial $f(x)$ in $\mathcal{K}$, suppose that some pair $(x, y = L^{-1}(z)) \in F_{2^n} \times F_{2^n}$ is a solution of the associated bivariate equation:

$$f(x) = y^{2^\delta} - y,$$

over $F_{2^n} \times F_{2^n}$. Then by regarding the signature value as $\sigma = x$ and hashed message value as $z = y^{2^\delta} - y$, we obtain:

$$f(\sigma) = z,$$

in the corresponding univariate representation of $MQ(2,n,n)$-trapdoor $F : F_2^n \to F_2^n$. In other words, $(L(y), x) = (z, \sigma)$ must be a valid message-signature pair of this

scheme, and this attack forms the existential forgery when successful. It can also be considered as a chosen message attack (Definition 39) against the scheme since the adversary restricts to a subset $Im(L)$ of the entire F_{2^n} the range for the messages z_j against which he expects to find a coincidence with some $f(\sigma_i)$.

It should be emphasized that the linearized binomial attacker must generate $\frac{T}{t}2^{\frac{n-\delta}{2}} > 2^{\frac{n-\delta}{2}}$ elements x's for their images $f(x)$'s in order to obtain the list

$$\{f(x_1), f(x_2), \ldots, f(x_{\frac{T}{t}2^{\frac{n-\delta}{2}}})\}$$

in which at least $2^{\frac{n-\delta}{2}}$ elements are expected to be inside $Im(L)$. Note also that since δ is guessed from $\{1, \ldots, n\}$, the total time complexity:

$$O(n \times \frac{T}{t}2^{\frac{n-\delta}{2}})$$

is necessary for the adversary so as to have the expected number of at least one coincidence in the two lists.

One of the advantages for the linearized binomial attack for this special class of D-O polynomials is that the linearized polynomial $y^{2^\delta} - y \in F_q[y]$ on the right-hand side is a *binomial*, and thus just guessing the values of δ allows us to *emulate* the D-O polynomial $f(x)$ on the left-hand side over the solution space. That is, randomly generated $2^{\frac{n-\delta}{2}}$ out of $\frac{T}{t}2^{\frac{n-\delta}{2}}$ elements $f(x_i) \in F_{2^n}$ are, indeed, expected to be gathered inside $Im(L)$ of size $2^{n-\delta}$. Since the extension degree n is always specified and fixed, this random guessing is, in fact, very efficient. (See the observation on δ later in the chapter. One can often regard δ as a small constant in many cases.) When this complexity $\frac{nT}{t}2^{\frac{n-\delta}{2}}$ is smaller than the birthday security parameter $\sqrt{\frac{\pi}{2}}2^{n/2}$ (Fact 2.27. [2]) the linearized binomial attack is more successful against the D-O polynomial $f(x)$ than normal birthday attack.

One interesting aspect of this approach is that we can analyze the *exact* number

of solutions of the bivariate equation from Theorem 34 by the *explicit* Weil sum values. To see this, recall that for the class of D-O polynomials in Theorem 34, we have the exact number t of the batches of the solutions. That is, for N we have:

$$N = (2^{\delta} + 1)t + 2^{\delta},$$

from Lemma 32 (t is the number of solution batches) and:

$$N = 2^n + (2^{\delta} - 1)S,$$

from Theorem 34 with Weil sum $S = \sum_{x \in F_{2^n}} \chi_1(\sum_{i=1}^{D} A_i x^{2^{s_i}+1})$. Therefore, we can now apply the Weil sum Algorithm 37 of Weil sum of the simplified D-O polynomial in order to obtain the exact values of S and t of $f(x)$.

Furthermore, in contrast to Gröbner basis attack, the complexity of the linearized binomial attack does not depend on the degree of the intermediate polynomial appearing in the computation. Instead, the computational complexity required by the attack here is closely related to the size of the greatest common divisor δ of the portions of exponents $s_1, \ldots, s_D$ and, as was shown in the simplified Weil sum algorithm 37, the dimension l of the kernel of the auxiliary linearized polynomial $T_D(x)$ in Theorem 34.

In sum, on the basis of the observation regarding the subjectivity of Dembowski-Ostrom polynomials to this attack, we can characterize the class of weak Dembowski-Ostrom polynomials in $\mathcal{K}$ under the emulation conditions in the following.

Theorem 42 *(Weak Dembowski-Ostrom Polynomial). Let F_{2^n} be a finite extension field of F_2 with extension degree n even. Let $\mathcal{K}$ be the class of Dembowski-Ostrom polynomial $f(x) = \sum_{i=1}^{D} A_i x^{2^{s_i}+1}$ over F_{2^n}, each of which satisfies the following emulation conditions: with $\delta = (s_1, \ldots, s_D, n)$*

1. *n/δ is even.*

2. *$\delta = (s_i, n)$ for each i,*

3. *s_i/δ is odd for each i, and*

4. *2δ divides $s_i - s_j$ for all $j \neq i$.*

And also let N be the number of solutions of bivariate equation $f(x) = y^\delta - y$ over $F_{2^n} \times F_{2^n}$ and suppose that the birthday security parameter is $\sqrt{\frac{\pi}{2}2^n}$. Then, if $f(x)$ is such that

$$N > 2^\delta + n\sqrt{\frac{2}{\pi}}2^{-\delta/2}(2^n - 1),$$

$f(x)$ is a weak Dembowski-Ostrom polynomial and is subject to the linearized binomial attack in time less than $\sqrt{\frac{\pi}{2}2^n}$.

As is previously mentioned, the classification of these weak polynomials involves the actual computation of Weil sum by Algorithm 37. In order to obtain N in the relation:

$$N = 2^n + (2^\delta - 1)S,$$

of Theorem 34, we need the actual computation of the Weil sum value:

$$S = \sum_{x \in F_{2^n}} \chi_1(\sum_{i=1}^{D} A_i x^{s_i+1}),$$

by the simplified Weil sum algorithm Algorithm 37 (which must resolves the sign as well). Nevertheless, it is also easily shown that in some particular combinations of the parameters used in the simplified Dembowski-Ostrom polynomial $f(x) = \sum_{i=1}^{D} A_i x^{2^{s_i}+1}$ of the emulation conditions Theorem 34 in $MQ(2, n, n)$-trapdoor, the linearized binomial attack can be *asymptotically* better than standard birthday attack.

Lemma 43 *Suppose $n \in N$ is even and arbitrarily fixed. Let F_{2^n} be a finite field of $MQ(2,n,n)$-trapdoor and $f(x) = \sum_{i=1}^{D} A_i x^{2^{s_i}+1}$ be a Dembowski-Ostrom polynomial satisfying the emulation conditions in Theorem 34 that represents $MQ(2,n,n)$-trapdoor. Then, for $1 \leq i \leq D$ there exists a combination of D and s_i such that:*

$$\delta = n/4.$$

Proof Consider the case $D = 2$. For every integer $i \geq 1$, define the even extension degree n of F_{2^n} over F_2 and (s_1, s_2): exponents of $f(x)$ such that:

$$\begin{cases} n = 4i, \\ s_1 = i, \\ s_2 = 3i. \end{cases}$$

It is easily shown that we have the valid emulation conditions:

$$\begin{cases} \delta = (s_1, n) = (s_2, n) = i, \\ n/\delta = 4 : \text{ even }, \\ s_1/\delta = 1 \text{ odd }, s_2/\delta = 3 \text{ odd and} \\ 2\delta = 2i \text{ divides } |s_2 - s_1|. \end{cases}$$

Therefore $\delta = n/4$. □

Moreover, we can obtain another theoretical estimate on the possible existence of the weak Dembowski-Ostrom polynomials under the linearized binomial attack, even when we take the Weil sum value S into consideration. To see this, recall that from Theorem 26, we have:

$$|S|^2 = 2^n \sum_{T_D(w)=0, w \in F_{2^n}} \chi_1(\sum_{i=1}^{D} A_i w^{2^{s_i}+1}).$$

From this we have:

$$|S| \leq 2^{\frac{n+l}{2}},$$

where $l = rank(ker(T_D))$ is the dimension of the kernel of the auxiliary linearized polynomial T_D. Suppose that we want to evaluate the relative sizes in the inequality comparing the complexities of the linearized binomial attack and birthday attack:

$$\frac{nT}{t}2^{\frac{n-\delta}{2}} < 2^{n/2},$$

(Note: for simplicity we omit $\sqrt{\pi/2} \approx 1.2533\cdots$). Since we have:

$$T = \frac{2^n - 1}{2^\delta + 1} \text{ and } t = \frac{1}{2^\delta + 1}\{2^n - 2^\delta + (2^\delta - 1)S\},$$

alternatively, we need to check if the inequality:

$$\frac{n(2^n - 1)}{2^n - 2^\delta + (2^\delta - 1)S} \cdot 2^{\frac{n-\delta}{2}} < 2^{n/2}$$

is satisfied. Thus, equivalently we want to check if:

$$S > \frac{1}{2^{\delta/2}(2^\delta - 1)}\{-(2^{\delta/2} - n)2^n - n + 2^{\frac{3\delta}{2}}\}.$$

It is easy to see that the right-hand side of this inequality is negative for reasonably large n, when we put, for example, $\delta = n/16 < 0.1n$ which is substantially smaller than n. That is, in this case *any* Dembowski-Ostrom polynomials $f(x)$ of the emulation conditions with *positive* Weil sum S on the left-hand side will surely satisfy the inequality. Of course, the value S is in between $\pm 2^{\frac{n+l}{2}}$ for $p = 2$ and is also dependent on the dimension l of the kernel of the auxiliary linearized polynomial $T_D(x)$ of $f(x)$.

However, while acknowledging the striking difference between the cases of $p = 2$ and odd prime p in regard to the proof methodology in explicit Weil sum evaluation [30, 31, 33], we have the results of Theorem 1.4 in [33] for F_{p^n} with odd p

implying that the Weil sum is:

$$S = \pm p^{\frac{n+l}{2}},$$

under the identical form and emulation conditions of simplified Dembowski-Ostrom polynomial $f(x) = \sum_{i=1}^{D} A_i x^{p^{s_i}+1}$ over F_{p^n}. That is, in odd p, the value of S can be either $p^{\frac{n+l}{2}}$ or $-p^{\frac{n+l}{2}}$. Since we know that δ can be as large as $n/4$ (Lemma 43), this observation implies the highly probable existence of the weak Dembowski-Ostrom polynomials under the linearized binomial attacks and if such polynomials exist with some $\delta = kn$, the attack complexity is less than $O(2^{n/2})$ by at least the factor of $2^{-\frac{kn}{2}}$.

Furthermore, it is also easily observed that $\delta = (s_1, \ldots, s_D, n)$ tends to remain small as D gets bigger. Therefore, in the time complexity $n \times \frac{T}{t} 2^{\frac{n-\delta}{2}}$ of linearized binomial attack we can often ignore the factor n by discretionarily setting δ from 1 and then working upward from there.

D. Concluding Remarks

Theorem 26 on the relation between the number of the solutions of the bivariate polynomial equations and the Weil sum values turned out to be crucial for the emulation of the Dembowski-Ostrom polynomials by the linearized binomials. Although the domain of weak Dembowski-Ostrom polynomials subject to the linearized binomial attacks are limited to those satisfying the emulation conditions, the cryptanalysis achieves asymptotic improvement of the birthday attacks on these polynomials. As Lemma 43 shows, the value δ can be as large as a quarter of the extension degree n.

It remains an open question to extend the emulation conditions on the oddness of n/δ with $p = 2$ proposed in Theorem 34 so that the scope of the linearized binomial attack might be further generalized.

CHAPTER VI

EXPERIMENT AND VARIATIONS

A. Introduction

In this chapter we will provide the simple experimental proof for the existence of some weakest type of Dembowski-Ostrom polynomials under the linearized binomial attack as Theorem 42. We also provide the brief specification of our implementation of the Weil sum Algorithm 37. We also consider some extensions of the Weak Dembowski-Ostrom polynomial into more general forms under the equivalence (by linear substitution of indeterminates).

B. Program Specification

The machine used for the experiments in this chapter is our personal computer which has a single Pentium 4 processor of 2GHz with 650 RAM. We installed a free portable C++ library *NTL* (A Library for doing Number Theory) by Shoup [51] for finite field operations in conjunction with *GMP*(the GNU Multi-Precision library [52]). We also used the public source codes of the textbook by Kreher and Stinson [53] for *gray code* generation algorithm.

There are two primary steps in the experiments. First, for simplicity, we generate all valid Dembowski-Ostrom polynomials in the simplified form (cf. 25) with the exponents satisfying the emulation conditions in Theorem 42 with coefficients 1 in F_{2^n}. More specifically, for a given (n, D), we assign the subset of $\mathcal{K}'$ of the Dembowski-Ostrom polynomials over F_{2^n} as:

$$\mathcal{K}' = \{\sum_{i=1}^{D} x^{2^{s_i}+1}, n/\delta \text{ even}, \delta = (s_i, n), s_i/\delta \text{ odd}, 2\delta \text{ divides } s_i - s_j\}.$$

In order to obtain all of the valid portions of exponents $\{s_1, \ldots, s_D\}$ under the emulation conditions (cf. Theorem 42), we generated all the possible D-subsets out of the integers $[0, n-1]$ and checked them against the emulation conditions.

Next, we apply the Weil sum Algorithm 37 for simplified Dembowski-Ostrom polynomials $f(x)$ in $\mathcal{K}'$. The following pseudo-code is the main portion of the Weil sum algorithm after computing the kernel of the auxiliary linearized polynomial of $f(x)$ in $\mathcal{K}'$ is determined with kernel basis `eta` in F_{2^n} of dimension `ker_dim`.

```
 \label{codes}


// Step 4. Gamma (Note A[i] = 1, case ker_dim > 0)
for(int i=0; i < D; i++)
    for(int j1=0; j1 < ker_dim; j1++)
        for (int j2=0; j2 < ker_dim; j2++) { // A[i] = 1
            gamma[i][j1][j2]
               = trace(eta[j1] * power(eta[j2],(long)exp2(s[i])));
        };
// End Step 4
// Step 5: Parity Checking
GF2 C; // parity in GF2 () for(int j= 0; j <(1 << ker_dim); j++){
    // Gray codes(Kreher and Stinson). Set coeff in Step 6.
    unsigned int T = GrayCodeUnrank(ker_dim, j);
    Get_gray_vecGF2(x, ker_dim, T);
    C = 0; // parity in GF2
    for(int i=0; i<D; i++){
        for(int j1=0; j1 < ker_dim; j1++){
```

```
            for (int j2=0; j2 < ker_dim; j2++){ // Loop. Step 6.
                // Note: Addition in GF2
                C += x[j1] * x[j2] * gamma[i][j1][j2];
            };
        };
    };
    U += IsOne(C); // Note: GF2 to ZZ by the method of class GF2E.
}; //End Step 5.
// Step 6: Sign Resolution
// Absolute value S_abs
S_abs = (long)exp2(n/2) * SqrRoot((long)exp2(ker_dim) - 2 * U);
if ((2*(1 + S_abs)) % ((long)exp2(delta)+1) == 0) // PLUS
   S = S_abs;
else if ((2*(1 - S_abs)) % ((long)exp2(delta)+1) == 0) // MINUS
   S = -S_abs;
return S;
//End Step 6.
```

Since the time complexity of Algorithm 37 is $O(Dl^2(n^3 + 2^l))$-time in terms of the binary operations on RAM, the most time-consuming portion is `Step 5` which goes through all the possible gray codes of dimension `ker_dim`. As was stated in Algorithm 30, however, no complex number computation is performed when computing the Weil sum `S`. Note also that the sign resolution for the Weil sum `S` with its absolute value `S_abs` in `Step 6` is quite simple, as shown in Theorem 36.

C. Existence of Weak Dembowski-Ostrom Polynomials

Since we have Lemma 43 regarding the size of δ and the asymptotic improvement by the linearized binomial attack, we made a simple simulation of the toy-sized example of the Dembowski-Ostrom polynomials $f(x)$ in $\mathcal{K}'$ for $D = 2$. The size of the extension degree of F_{2^n} is taken up to $n = 24$. In this example, the combination of parameters in Lemma 43 is expected to give the improved efficiency of linearized binomial attack on $f(x)$ by factor of $2^{\delta/2} = 2^{n/8}$.

Table I. The Parameters of Weak Dembowski-Ostrom Polynomials ($D = 2, \delta = n/4$).

n	8	12	16	20	24
δ	2	3	4	5	6
(s_1, s_2)	(2, 6)	(3,9)	(4,12)	(5, 15)	(6, 18)
$[x^d]$	16,1,1,16	64,1,1,64	256,1,1,256	1024,1,1,1024	4096,1,1,4096
N	1,024	32,768	1,048,576	33,554,432	1,073,741,824
N_{low}	817	13,868	209,151	2,957,690	40,154,700
S	256	4,096	65,536	1,048,576	16,777,216

The row $[x^d]$ in the above table indicates all of the tuples of the degrees: $(2^{2_1}, 2^1, 2^{s_1+s_2}, 2^{s_1+y_2})$ of the four monomials originally assigned in the auxiliary linearized polynomial:

$$T_2(x) = x^{2^{2s_1}} + x + x^{2^{s_1+s_2}} + x^{2^{s_1+y_2}}.$$

The row N_{low} is the integer portion of the right-hand side of the identity given in Theorem 42, i.e., the actual value:

$$2^\delta + n\sqrt{\frac{2}{\pi}}2^{-\delta/2}(2^n - 1),$$

with the δ and n in the same column in the table.

The above table shows that even though we set all the coefficients 1 in $f(x) = x^{2^{s_1}+1} + x^{2^{s_2}+1}$, the Dembowski-Ostrom polynomials with the combination of parameters in Lemma 43 are always weak against the linearized binomial attack criteria in Theorem 42. It is easily expected that when D gets larger, δ may be restrained to be small with respect to n. Therefore the case in the table can be considered as one of the most extreme cases of the security of the Dembowski-Ostrom polynomials, whereby the largest asymptotic gain can be achieved by the linearized binomial attack.

It is also interesting to note that all the auxiliary linearized polynomial in the table have:

$$T_2(x) = x^{2^{2s_1}} + x + x^{2^{s_1+s_2}} + x^{2^{s_1+y_2}} = 0.$$

That is, each $T_2(x)$ is a zero polynomial. This situation occurs simply because we assumed $A_1 = A_2 = 1$ in F_{2^n} and for this quite sparse case $D = 2$, all of the four monomials assigned in $T_2(x)$ occasionally got *paired* with each other (see the row x^d in the table) and canceled out the coefficients with arithmetic $1 + 1 = 0 \in F_{2^n}$. Consequently, the kernel of $T_2(x)$ is the entire F_{2^n} itself so that we had to loop `Step 5` in the parity checking portion for all possible gray codes of dimension `ker_dim` $= n$. Therefore, it is easily observed that if we assign the general values in F_{2^n} to the coefficients A_i this cancelation should be infrequent. So the size l should tend to be much smaller than n and it will lead to the better efficiency in loop Step 5.

We explicitly list the weak Dembowski-Ostrom polynomials with $\delta = n/4$ in Table C.

- For $n = 8$, $f(x) = x^{2^2+1} + x^{2^6+1} = x^5 + x^{65}$ in $F_{2^8}[x]$.
- For $n = 12$, $f(x) = x^{2^3+1} + x^{2^9+1} = x^9 + x^{513}$ in $F_{2^{12}}[x]$.
- For $n = 16$, $f(x) = x^{2^4+1} + x^{2^{12}+1} = x^{17} + x^{4097}$ in $F_{2^{16}}[x]$.

- For $n = 20$, $f(x) = x^{2^5+1} + x^{2^{15}+1} = x^{33} + x^{32769}$ in $F_{2^{20}}[x]$.
- For $n = 24$, $f(x) = x^{2^6+1} + x^{2^{18}+1} = x^{65} + x^{262145}$ in $F_{2^{24}}[x]$.

We conjecture from the above that the simplified D-O polynomials defined as:

$$f(x) = x^{2^{n/4}+1} + x^{2^{3n/4}+1} \in F_{2^n}[x] \tag{6.1}$$

with $n = 4i$, $i \geq 2$ from Lemma 43 form an *infinite* series of weak D-O polynomials of Theorem 42.

D. Variations of Weak Dembowski-Ostrom Polynomials

For any linearized polynomial $L(x) \in F_{2^n}[x]$ and any D-O polynomial $f(x) \in F_{2^n}[x]$ the compositions $L \circ f(x)$ and $f \circ L(x)$ are D-O polynomials. It is easy to see that since there are $\prod_{i=0}^{n-1}(2^n - 2^i)$ invertible $n \times n$-matrices over F_2, there are the same number of the corresponding invertible linearized polynomials in $F_{2^n}[x]$. In addition, the reduction of D-O polynomial in $F_{2^n}[x]$ by $x^{2^n} - x$ is again a D-O polynomial and the number of the solutions of the equations is identical. I.e., given $f(x)$ an arbitrary D-O polynomial and let $\overline{f}(x)$ be a reduction of $f(x)$ by $x^{2^n} - x$, then for some $q(x), \overline{f}(x) \in F_2^n[x]$ we have $f(x) = q(x)(x^{2^n} - x) + \overline{f}(x)$ where $deg(\overline{f}) \leq 2^n - 1$ and thus we have: for $(x, z) \in F_{2^n} \times F_{2^n}$ $f(x) = z$ iff $\overline{f}(x) = z$.

If one could take some invertible affine 2-polynomial $A(x) = L(x) + b \in F_{2^n}[x]$ (Definition 3.54 [19]), compositions and reduction above may lead to some (central) polynomial $f(x) = \sum_{i=1}^{D} a_i x^{2^{\alpha_i}+2^{\beta_i}} + \sum_{j=1}^{L} b_j x^{2^{\gamma_i}} + c$, $(D, L \in N$, $a_i, b_j, c \in F_{2^n}$, $\alpha_i \leq \beta_i$, $2^{\alpha_i} + 2^{\beta_i}, 2^{\gamma_i} \leq 2^n - 1)$. More explicitly by using (6.1) we obtain the *equivalent* polynomials

$$\overline{f}(x) \;=\; A(x)^{2^{n/4}+1} + A(x)^{2^{3n/4}+1} \mod x^{2^n} - x, \text{ or} \tag{6.2}$$

$$\overline{f}(x) \;=\; A(x^{2^{n/4}+1} + x^{2^{3n/4}+1}) \mod x^{2^n} - x \tag{6.3}$$

with an arbitrary invertible affine 2-polynomials $A(x) \in F_{2^n}[x]$. It is easy to see that we can input this $\overline{f}(x)$ in linearized binomial attack of Definition 41 and $N(\overline{f}, y^{2^\delta} - y) = N(f, y^{2^\delta} - y)$ still holds. Therefore, we can regard $\overline{f}(x)$ as a weak polynomial according to Theorem 42. (The use of Formula (6.3) needs more care when emulating by $y^{2^n} - y$ because $A(x)$ is composed to $f(x)$ from the left.)

Note that these polynomials were neither explicitly known nor evaluated its security by any other previous multivariate quadratic schemes in the literature including Gröbner basis methods [6]. Thus, the precise complexity analysis of other methods including Gröbner basis methods is left as an interesting open problem. For instance, the performance of the Gröbner basis method [6] on the HFE system could differ from the randomly generated MQ problems mainly because HFE could take degree less than some predetermined upper bound. On the other hand, the degrees of the polynomials of (6.2) or (6.3) can varies according to an composed invertible affine 2-polynomial $A(x)$. In fact, we may randomly choose a out of $\prod_{i=0}^{n-1}(2^n - 2^i) + 2^n$ invertible affine 2-polynomials, and some of them can have the shape of HFE polynomials with proper choice of $A(x)$. Therefore, in certain specific cases Gröbner basis methods might perform merely exponentially in extension degree n of F_2 against the corresponding systems of quadratic equations.

As a result, when multivariate quadratic schemes (such as STS and variations: [10, 11, 12]. UOV and others: [13, 14, 9]) are attempting to provide "short" signature schemes, the security under birthday attack or customized methods such as linearized binomial attack (Definition 41) becomes more crucial. Therefore, we must carefully examine the potential existence of weak polynomials as partially shown in this paper in addition to the security under other potential (algebraic) attacks. Since our

results indicate that these weak D-O polynomials and their equivalent forms might exist regardless of size n, any MQ-based short signatures should explicitly eliminate such weak polynomial instances from its key generation algorithms.

Another open question, beside the conjecture of the existence of infinite series of (6.1) is to enumerate all the possible weak D-O polynomials defined by Theorem 42 and obtain their equivalent forms under all possible invertible affine 2-polynomials (with reduction by $x^{2^n}-x$). For $D > 2$ one expects the value δ may tend to be smaller, but we may still obtain the linearized binomial attacks of asymptotic improvements by $2^{\delta/2}$.

E. Concluding Remarks

We implemented the Weil sum Algorithm 37 and empirically showed the existence of the class of weak Dembowski-Ostrom polynomial characterized in Theorem 42 under the linearized binomial attack. Although it is still an open question whether the Table C can be extended into the infinitely many large n, we complete the proof of existence of the class of weak Dembowski-Ostrom polynomials in Theorem 42 in Chapter V by performing a simulation of the most extreme cases from Lemma 43. In this case, the linearized binomial attack is asymptotically better than the generic birthday attack on the Dembowski-Ostrom polynomials under these conditions by factor of at least $2^{n/8}$.

We also described the possible variations of the weak polynomials under the equivalence relations and provided some observations of the security of these polynomials under the other attacks such Gröbner basis methods. Although the experiment were not meant to be exhaustive, we could partially demonstrated by simulation that the weak Dembowski-Ostrom polynomials of formula (6.1) are likely to exist for n of

arbitrary size.

CHAPTER VII

CONCLUSION

We proposed a new cryptanalytic application of Weil sum by developing the explicit Weil sum method, its algorithm and the formulas for the number of solutions of the associate bivariate equations. The customized birthday attack in this paper proposes a new way to solve the special subclass of the system of multivariate quadratic equations by using the univariate polynomial representation. The analysis can be done in the univariate representation, but the method is applicable to the corresponding multivariate mapping $F_2^n \to F_2^n$. The principle behind this analysis is fundamentally different from those by Gröbner basis computation (cf. [6]). Therefore, the application of Weil sums of this type for solving systems of multivariate equations may be of independent interest.

The parity checking-styled Weil sum Algorithm 30 in Chapter III can avoid the complex number calculation for finite fields of characteristic $p = 2$. The algorithm computes the absolute values of the Weil sums of the generic univariate polynomials which fully characterize MQ problem of n polynomials in n indeterminates over F_2. In Chapter IV, we developed a theorem which relates the Weil sum value to the number of solutions of the bivariate equation in Theorem 34. The sign resolution in Theorem 36 contrasts to that of others when p is odd and enabled us to obtain the concrete value of the Weil sum of the Dembowski-Ostrom polynomials under the emulation conditions in Algorithm 37.

A new attack called linearized binomial attack was developed in Definition 41 which characterizes the weak Dembowski-Ostrom polynomials in MQ problem (Theorem 42). Although the domain of weak Dembowski-Ostrom polynomials under the linearized binomial attacks are limited to the emulation conditions, the cryptanalysis

can achieve an asymptotic improvement from the birthday attacks on these polynomials. A simple case in Lemma 43 shows that the value δ can be as large as a quarter of the extension degree n. Also, unlike the previously known algebraic attacks including those based on the Gröbner basis algorithms, the time complexity does not depend on the degree of the polynomials.

In Chapter VI, we complete the proof of existence of the class of weak Dembowski-Ostrom polynomials in Theorem 42 by performing a simple computer simulation of the weakest Dembowski-Ostrom polynomials of Lemma 43. In this case, the linearized binomial attack is asymptotically better than the generic birthday attack on the Dembowski-Ostrom polynomials under these conditions by factor of at least $2^{n/8}$.

In the case when central mapping or polynomial is not a permutation polynomial the number of solutions of equation $f(x) = z$ over F_{2^n} has a certain probability distribution (cf. Theorem 6.16 and 6.17 [19]). Therefore, the *information* regarding the number of the solutions of the generated equations $f(x) = z$ at each step in collision search may allow adversaries to mount some sort of *adaptive attack* (Section 11.2.4. [2]). For example, if $f(x)$ is a weak DO polynomial or its equivalent polynomial then the adversary may adaptively restrict the generation of (hashed) message in F_{2^n} into a subset of the entire range for the messages z against which he expects to more easily find a coincidence with some $f(x)$. That is, such adaptive attacks may deliberately select $z \in F_{2^n}$ of larger $|f^{-1}(z)|$ so that the random selection of $x \in F_{2^n}$ could more likely satisfy $f(x) = z$.

One of the design principles that this paper indicates is that one should select the MQ trapdoor of high balance [54] in order to resist birthday attack. Interestingly from this point of view, the permutation monomial $x^{2^\alpha+1}$ over F_{2^n} with $\gcd(\alpha, n) = 1$ is of balance 1 so in principle, the complexity of birthday attack should follow the common notion of square root complexity $O(2^{n/2})$. Coulter et al. [29] discovered a

new class of D-O permutation polynomials. Thus, it is an open question if one can build a new extension of MI schemes with these permutation polynomials.

Although we handled only $p = 2$ of the base field F_2, the weak D-O polynomials such as in Table C are expected to exist for *infinitely many* n. And also, the birthday paradox techniques used in this paper are quite general and widely applicable in many other cryptanalytic problems. Therefore, key generation algorithms for multivariate quadratic cryptosystems should explicitly deal with these weak D-O polynomials and their equivalent instances in the design specifications, especially for *short* signature schemes.

REFERENCES

[1] Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput.*, 26(5):1484–1509, 1997.

[2] Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone. *Handbook of Applied Cryptography*. CRC Press, 1997.

[3] Bart Preneel et al. NESSIE D20-NESSIE security report. A Draft version 0.15 (beta), 2004. http://citeseer.ist.psu.edu/549422.html.

[4] Michael R. Garey and David S. Johnson. *Computer and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.

[5] Jacques Patarin. Hidden fields equations (HFE) and isomorphisms of polynomials (IP): Two new families of asymmetric algorithms. In *Proceedings of Advances in Cryptology - EUROCRYPT '96*, volume 1070 of *Lecture Notes in Computer Science*, pages 33–48. Springer, 1996.

[6] Jean-Charles Faugère and Antoine Joux. Algebraic cryptanalysis of hidden field equation (HFE) cryptosystems using Gröbner bases. In *Proceedings of Advances in Cryptology - CRYPTO 2003*, volume 2729 of *Lecture Notes in Computer Science*, pages 44–60. Springer, 2003.

[7] Tsutomu Matsumoto and Hideki Imai. Public quadratic polynominal-tuples for efficient signature-verification and message-encryption. In *Proceedings of Advances in Cryptology - EUROCRYPT '88*, volume 330 of *Lecture Notes in Computer Science*, pages 419–453. Springer, 1988.

[8] Jintai Ding and Dieter Schmidt. Cryptanalysis of HFEv and internal perturbation of HFE. In *Proceedings of Public Key Cryptography - PKC 2005*, volume 3386 of *Lecture Notes in Computer Science*, pages 288–301. Springer, 2005.

[9] Jacques Patarin, Louis Goubin, and Nicolas Courtois. C^*_{-+} and HM: Variations around two schemes of T. Matsumoto and H. Imai. In *Proceedings of Advances in Cryptology - ASIACRYPT '98*, volume 1514 of *Lecture Notes in Computer Science*, pages 35–49. Springer, 1998.

[10] Tzuong-Tsieng Moh. A public key system with signature and master key functions. *Communications in Algebra*, 27(5):2207–2222, 1999.

[11] Jiun-Ming Chen and Bo-Yin Yang. A more secure and efficacious TTS signature scheme. In *Proceedings of Information Security and Cryptology - ICISC 2003*, volume 3574 of *Lecture Notes in Computer Science*, pages 320–338. Springer, 2003.

[12] Louis Goubin and Nicolas Courtois. Cryptanalysis of the TTM cryptosystem. In *Proceedings of Advances in Cryptology - ASIACRYPT 2000*, volume 1976 of *Lecture Notes in Computer Science*, pages 44–57. Springer, 2000.

[13] Aviad Kipnis, Jacques Patarin, and Louis Goubin. Unbalanced Oil and Vinegar signature schemes. In *Proceedings of Advances in Cryptology - EUROCRYPT '99*, volume 1592 of *Lecture Notes in Computer Science*, pages 206–222. Springer, 1999.

[14] Jacques Patarin, Louis Goubin, and Nicolas Courtois. Improved algorithms for Isomorphisms of Polynomials. In *Proceedings of Advances in Cryptology - EUROCRYPT '98*, volume 1403 of *Lecture Notes in Computer Science*, pages 184–200. Springer, 1998.

[15] Jacques Patarin and Louis Goubin. Asymmetric cryptography with S-boxes. In *Proceedings of Information Security and Cryptology - ICISC '97*, volume 1334 of *Lecture Notes in Computer Science*, pages 369–380. Springer, 1997.

[16] National Institute of Standards and U.S. Department of Commerce Technology. FIPS PUB 197, Advanced Encryption Standard (AES), 2001. http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf.

[17] Joan Daemen and Vincent Rijmen. *The Design of Rijndael: AES—the advanced encryption standard*. Springer-Verlag, 2002.

[18] Jacques Patarin. Cryptoanalysis of the Matsumoto and Imai public key scheme of Eurocrypt'88. In *Proceedings of Advances in Cryptology - CRYPTO '95*, volume 963 of *Lecture Notes in Computer Science*, pages 248–261. Springer, 1995.

[19] Rudolf Lidl and Harald Niederreiter. *Finite Fields*, volume 20 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, Cambridge, second edition, 1997.

[20] Nicolas T. Courtois. Short signatures, provable security, generic attacks and computational security of multivariate polynomial schemes such as HFE, Quartz and Sflash. Cryptology ePrint Archive, Report 2004/143, 2004. http://eprint.iacr.org/.

[21] Jacques Patarin, Nicolas Courtois, and Louis Goubin. QUARTZ, 128-bit long digital signatures. In *Proceedings of Topics in Cryptology - CT-RSA 2001*, volume 2020 of *Lecture Notes in Computer Science*, pages 282–297. Springer, 2001.

[22] Jean-Sébastien Coron (Editor). Hardness of the main computational problems used in cryptography. D.AZTEC-4, ECRYPT public documents, 2005.

http://www.ecrypt.eu.org/documents/D.AZTEC.4-1.1.pdf.

[23] Jean-Charles Faugère. A new efficient algorithm for computing Gröbner bases (F_4). *Journal of Pure and Applied Algebra*, 139(1-3):61–88, 1999.

[24] Jean-Charles Faugère. A new efficient algorithm for computing Gröbner bases without reduction to zero F_5. In *Proceedings of International Symposium on Symbolic and Algebraic Computation - ISSAC 2002*, pages 75–83. ACM Press, 2002.

[25] Nicolas Courtois, Magnus Daum, and Patrick Felke. On the security of HFE, HFEv- and Quartz. In *Proceedings of Public Key Cryptography - PKC 2003*, volume 2567 of *Lecture Notes in Computer Science*, pages 337–350. Springer, 2003.

[26] Mehdi-Laurent Akkar, Nicolas Courtois, Romain Duteuil, and Louis Goubin. A fast and secure implementation of Sflash. In *Proceedings of Public Key Cryptography - PKC 2003*, volume 2567 of *Lecture Notes in Computer Science*, pages 267–278. Springer, 2003.

[27] Nicolas T. Courtois, Louis Goubin, and Jacques Patarin. SFLASHv3, a fast asymmetric signature scheme. Cryptology ePrint Archive, Report 2003/211, 2003. http://eprint.iacr.org/.

[28] Peter Dembowski and Thedore G. Ostrom. Planes of order n with collineation groups of order n^2. *Math Z*, 103:239–258, 1968.

[29] Aart Blokhuis, Robert S. Coulter, Marie Henderson, and Christine M. O'Keefe. Permutations amongst the Dembowski-Ostrom polynomials. In *Proceedings of*

The Fifth International Conference on Finite Fields and Applications Fq5 (Augsburg 1999). Springer, 2001.

[30] Robert S. Coulter. Explicit evaluations of some weil sums. *Acta Arithmetica*, 83:241–251, 1998.

[31] Robert S. Coulter. Further evaluations of weil sums. *Acta Arithmetica*, 86:217–226, 1998.

[32] Robert S. Coulter. On the evaluation of a class of Weil sums in characteristic 2. *NZ J. Mathematics*, 28(2):171–184, 1999.

[33] Donald Mills. On the evaluation of Weil sums of Dembowski-Ostrom polynomials. *Journal of Number Theory*, 92(1):87–98, 2002.

[34] Robert S. Coulter and Marie Henderson. The compositional inverse of a class of permutation polynomials over a finite field. *Bull. Austral. Math. Soc.*, 65:521–526, 2002.

[35] Hans Dobbertin. Almost perfect nonlinear power functions on GF(2^n): The Welch case. *IEEE Transactions on Information Theory*, 45(4):1271–1275, 1999.

[36] Nicolas Courtois. The security of hidden field equations (HFE). In *Proceedings of Topics in Cryptology - CT-RSA 2001*, volume 2020 of *Lecture Notes in Computer Science*, pages 266–281. Springer, 2001.

[37] Nicolas Courtois, Louis Goubin, Willi Meier, and Jean-Daniel Tacier. Solving underdefined systems of multivariate quadratic equations. In *Proceedings of Public Key Cryptography - PKC 2002*, volume 2274 of *Lecture Notes in Computer Science*, pages 211–227. Springer, 2002.

[38] Aviad Kipnis and Adi Shamir. Cryptanalysis of the HFE public key cryptosystem by relinearization. In *Proceedings of Advances in Cryptology - CRYPTO '99*, volume 1666 of *Lecture Notes in Computer Science*, pages 19–30. Springer, 1999.

[39] Henri Gilbert and Marine Minier. Cryptanalysis of SFLASH. In *Proceedings of Advances in Cryptology - EUROCRYPT 2002*, volume 2332 of *Lecture Notes in Computer Science*, pages 288–298. Springer, 2002.

[40] Robert S. Coulter, George Havas, and Marie Henderson. Giesbrecht's algorithm, the HFE cryptosystem and Ore's p^s-polynomials. In *Proceeding of Computer Mathematics: ASCM 2001*, volume 9 of *Lecture Notes Series on Computing*, pages 36–45. World Scientific, 2001.

[41] Pierre-Alain Fouque, Louis Granboulan, and Jacques Stern. Differential cryptanalysis for multivariate schemes. In *Proceedings of Advances in Cryptology - EUROCRYPT 2005*, volume 3494 of *Lecture Notes in Computer Science*, pages 341–353. Springer, 2005.

[42] Christopher Wolf. Efficient public key generation for HFE and variations. In *Cryptographic Algorithms and their Uses*, pages 78–93. Queensland University of Technology, 2004.

[43] Christopher Wolf and Bart Preneel. Large superfluous keys in multivariate quadratic asymmetric systems. In *Proceedings of Public Key Cryptography - PKC 2005*, volume 3386 of *Lecture Notes in Computer Science*, pages 275–287. Springer, 2005.

[44] Dan Boneh, Glenn Durfee, and Yair Frankel. An attack on RSA given a small fraction of the private key bits. In *Proceedings of Advances in Cryptology -*

ASIACRYPT '98, volume 1514 of *Lecture Notes in Computer Science*, pages 25–34. Springer, 1998.

[45] Dan Boneh and Ramarathnam Venkatesan. Hardness of computing the most significant bits of secret keys in Diffie-Hellman and related schemes. In *Proceedings of Advances in Cryptology - CRYPTO '96*, volume 1109 of *Lecture Notes in Computer Science*, pages 129–142. Springer, 1996.

[46] Igor E. Shparlinski. Playing "hide-and-seek" in finite fields: The hidden number problem and its applications. In *Proceedings of 7th Spanish Meeting on Cryptology and Information Security*, volume 1, pages 49–72. Univ. of Oviedo, 2002.

[47] Maria Isabel Gonzalez Vasco, Mats Naslund, and Igor Shparlinski. The hidden number problem in extension fields and its applications. In *Proceedings of LATIN 2002: Theoretical Informatics*, volume 2286 of *Lecture Notes in Computer Science*, pages 105–117. Springer, 2002.

[48] Tomohiro Harayama. On the weil sum evaluation of central polynomial in multivariate quadratic cryptosystem. Cryptology ePrint Archive, Report 2006/075, 2006. http://eprint.iacr.org/2006/075.

[49] Tomohiro Harayama and Donald K. Friesen. Weil sum for birthday attack in multivariate quadratic cryptosystem. *Journal of Mathematical Cryptology*, 1(1):79–104, 2007.

[50] Victor Shoup. *A Computational Introduction to Number Theory and Algebra*. Cambridge University Press, 2005. http://shoup.net/ntb/.

[51] Victor Shoup. NTL: A library for doing Number Theory. Version 5.4., 2005. http://shoup.net/ntl/.

[52] Torbjörn Granlund et al. GNU multiple precision arithmetic library 4.1.4., 2004. http://swox.com/gmp/.

[53] Donald. L. Kreher and Douglas. R. Stinson. *Combinatorial Algorithms: Generation, Enumeration, and Search*. CRC Press, 1999.

[54] Mihir Bellare and Tadayoshi Kohno. Hash function balance and its impact on birthday attacks. In *Proceedings of Advances in Cryptology - EUROCRYPT 2004*, volume 3027 of *Lecture Notes in Computer Science*, pages 401–418. Springer, 2004.

VITA

Tomohiro Harayama was born in Osaka, Japan, the son of Tatsuo Harayama and Mieko Harayama. He entered Kyoto University in Kyoto, Japan in 1993. After receiving the degree of Bachelor of Science in 1998, he entered Japan Advanced Institute of Science and Technology in Ishikawa, Japan where he received the degree of Master of Science in 2000. He entered Texas A&M University in August 2000 and received his Ph.D. in Computer Science in May 2007. He is employed as an Expert Researcher at Information Security Research Center, National Institute of Information and Communications Technology in Tokyo, Japan since June 2006.

The typist for this disseration was Tomohiro Harayama.